AD-A282 105

An Introduction to Variable-Resolution Modeling and Cross-Resolution Model Connection

Paul K. Davis

DTIC
SELECTED
JUL 25 1994
S
B
D
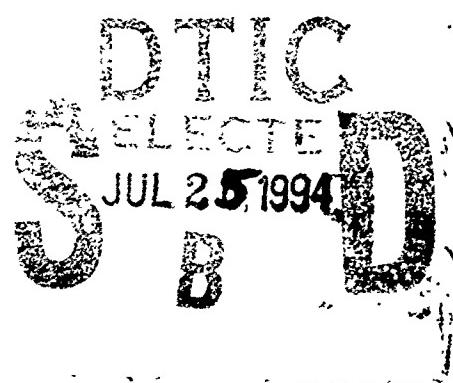
COPY 94-23138
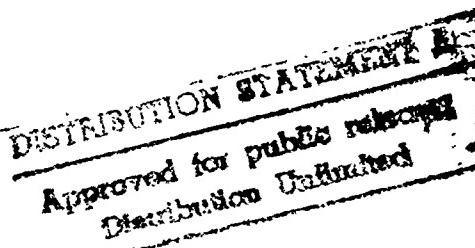
DTIC QUALITY INSPECTED 1

94 7 22 217

RAND / NATIONAL DEFENSE RESEARCH INSTITUTE

R-4252-DARPA

An Introduction to Variable-Resolution Modeling and Cross-Resolution Model Connection

Paul K. Davis

Prepared for the
Defense Advanced Research Projects Agency

RAND

PREFACE

This study was prepared for the Defense Advanced Research Projects Agency. The work was performed in the Applied Science and Technology program of RAND's National Defense Research Institute (NDRI), a federally funded research and development center sponsored by the Office of the Secretary of Defense and Joint Staff. It is the introductory piece of a trilogy of papers working through simple examples to illustrate a multitude of issues that arise in variable-resolution modeling. The other papers are: Richard J. Hillestad and Mario L. Juncosa, *Cutting Some Trees to See the Forest: On Aggregation and Disaggregation in Combat Models*, RAND R-4250-DARPA, and Richard J. Hillestad, John Owen, and Donald Blumenthal, *Experiments in Variable-Resolution Combat Modeling*, RAND N-3631-DARPA. Initial versions of all three papers were presented at a conference on variable-resolution modeling organized by RAND and the University of Arizona, which was sponsored by DARPA and the Defense Modeling and Simulation Office in May of 1992.

SUMMARY

It 's commonly necessary in analysis and other activities involving models to work at two or more levels of resolution. Resolution is a relative concept, but at any given level we may think of increasing resolution (e.g., to gain greater insight about phenomenology) or decreasing resolution (e.g., to produce something more comprehensible and more appropriate for policy analysis or operational decision support). Sometimes we can calibrate lower-resolution models with higher-resolution models and, to some extent, vice versa.

Ideally, models (or integrated model families) would be built from the outset with variable-resolution capability. Often, however, we find ourselves having to do cross-resolution work by linking existing models that were not designed to be connected. Although there are software techniques for connecting "dissimilar models," these techniques do not guarantee that it is substantively meaningful to do so. Similarly, while many workers have found off-line methods for tuning one model of an alleged family to be at least somewhat consistent with another model of the family, the consistency is sometimes more apparent than real. To put it differently, taking existing models with varied resolutions and declaring them to constitute a hierarchical family is sometimes quite misleading because the models are neither integrated nor readily integratable. Further, even when means for relating the models sensibly have been developed (either with off-line methods or model-connection methods), doing so may involve complex, tedious, error-prone, and expensive calibration efforts.

This study describes building models with variable-resolution capability. It also describes generic substantive challenges in connecting models developed independently, and it recommends that such model-connection activities be guided by design work to identify how the models would have been developed in the first place if their subsequent integration had been a goal. This approach can make it possible to connect the models usefully and comprehensively with one set of adaptations rather than a series of incremental patches. In other cases the approach may convince users to commission the building of a new variable-resolution model by demonstrating that the existing models will never work together well.

The study emphasizes a particular approach called integrated hierarchical variable-resolution modeling (IHVR), in which critical processes (not merely objects) are designed hierarchically, with clear and

meaningful relationships among levels so that we can move up and down a given process hierarchy without mental disruption. IHVR design makes it possible, at each node of a hierarchy, to provide the option to either generate variables from the higher-resolution processes farther down the hierarchical tree or to specify those variables as parameters. A good IHVR design should also specify how to calibrate those parameters by conducting experiments with the higher-resolution processes and performing appropriate statistical averages across cases and time. This is in contrast to the common approach in combat modeling of attempting to make models work together by "tuning" parameters in unnatural ways that make little sense phenomenologically or mathematically, and of basing such "tuning" on allegedly representative cases rather than a well-defined mathematical averaging process. While rigorous IHVR designs are not always possible (e.g., because aggregation does not always work well and because some processes interact strongly), it is often possible to develop approxima-tions that create useful hierarchies, if we know to try. The approximations may be valid in only certain domains (e.g., domains relatively close to certain calibration points or for times short compared with the time at which the system changes substantially in particular respects), but that may nonetheless be quite useful.

Much of the study is an attempt to provide a synthetic case history of how failure to design for variable resolution leads to confusion and trouble and how IHVR methods can improve the situation. The case history involves attempting to connect two ground-combat models that are simple enough to describe in detail. Although both of them are actually highly simplified in absolute terms, we consider one of them, for the purposes of this study, to be simple and aggregate and the other to be detailed and higher in resolution. This case history demonstrates basic issues and illustrates generic problems such as the confusion that exists when models use the same terms for different concepts or have different perspectives on the same issue. The case history is completed by sketching IHVR methods by which the models can be made more consistent or perhaps even integrated. Abstracting from these examples produces guidelines for those approaching model-connection challenges or those beginning to design models or model families for variable resolution. Perhaps the most important principle is that there needs to be greater emphasis on design and graphical depictions of those designs, along with naming conventions that clarify relationships across levels. While rapid prototyping is highly desirable, and it is very important to maintain flexibility to adjust models as we gain experience, a moderate amount of

initial design work (measured in weeks, not months) with IHVR concepts in mind can more than pay for itself in subsequent coherence.

A corollary here is that the Department of Defense should be cautious in commissioning efforts to connect large and complex models because of assumed benefits in being able to cross levels of resolution. Such efforts should be preceded by studies that review model designs and address the kinds of issues discussed in this study. Such studies will sometimes conclude that it makes no sense to connect particular models, whether or not connecting them is possible. Such studies could have major ramifications for advanced distributed-simulation programs and for efforts to increase interoperability and reusability of models.

ACKNOWLEDGMENTS

CONTENTS

FIGURES

TABLES

1. INTRODUCTION

This study is an introduction to the subject of variable-resolution modeling (VRM) and the closely related issue of developing integrated families of models with varied resolutions. It addresses the following questions: (1) What *is* variable-resolution modeling? (2) Why might one want it? (3) What forms can it take and how does it relate to "families of models"? and (4) How should one go about it? After providing definitions and some basic concepts,[1] I work through an extremely simple but concrete combat-modeling problem to illustrate generic issues. A theme here is that the usual approach to model building, even if undertaken professionally, results in a diversity of independent models with different but overlapping resolutions, models that are difficult to use together because of both obvious and subtle differences in perspective, assumption, and definition. To move across levels of resolution readily it is highly desirable to have designed for that in the first place, or to pay the price of redesigning existing models so that they are truly integrated. Lashing models together without such redesign is likely to cause trouble or require substantial experience, skill, and time on the part of the analyst. Lashups may run, but using them efficiently and understanding the results is a different matter. After illustrating issues and some general methods using the simple models, I conclude with suggestions for both designing from scratch and redesigning existing models to integrate them in a family.

---

[1]See also Appendix A, which provides background for the study and gives citations to some of the more relevant academic literature.

2. DEFINITIONS AND BASIC CONCEPTS

DEFINITIONS

To proceed we need some definitions. The most important for the purposes of this paper are:

Variable-resolution modeling: building models or model families so that users can change readily the resolution at which phenomena are treated (either by "turning resolution knobs" within a single model or turning from one model to another in a family).

Cross-resolution model connection: linking existing models with different resolutions (the linkage may be in software so that the models operate together or "external" in which case outputs of one are transferred manually to become inputs of another).

Seamless design: permiting changing resolution with (a) smooth consistency of representation (description) and (b) consistency of prediction.

The distinction here is that the term "variable-resolution modeling" applies to designing a *new* model (or family of models). "Cross-resolution model connection" applies when talking about combining *existing* models of different resolution, models that were not originally designed to be combined. Cross-resolution work is often a matter of "coping." "Seamlessness" in this context means that, when we change resolution, either within a single model or by moving from one model to another within a family, we can do so without mental disruptions and with some confidence that the results will be consistent in a sense to be discussed later. While we cannot aspire to continous variable resolution analogous to the zooming of a camera, we *can* aspire to models that allow us to make graduated changes in resolution that are easy to follow and, within limits, valid.

The reason for discussing variable-resolution modeling and cross-resolution model connection together is that the latter can often be accomplished best if we step back and pretend to have the luxury of starting over again. After understanding what we would *like* to have, we can then make a set of reasonable model adaptations once and for all, rather than start a process of sequential patches.

SUBTLETIES IN THE CONCEPT OF RESOLUTION

"Resolution" is usually treated as a primitive concept, but it is in fact rather subtle, as Fig. 1 suggests. Indeed, we use the same word for very different concepts. For example:

1. A model may have higher resolution because it deals with more fine-grained entities (e.g., companies rather than battalions).[1]

2. A model may have higher resolution than another with the same entities because it ascribes to those entities a richer set of attributes. For example, the companies in one model may be characterized by firepower and in another model by detailed weapon holdings. Or, to use a different example, targets may be described as having a wavelength-dependent spectral radiance rather than a mere brightness.

3. A model with the same entities and attributes as another may have higher resolution because it describes the relationships among those attributes in more detail (called logical dependencies here). For example, one model may describe the spatial relationship among ground-combat units in rich detail, depending on cir-

RAND#100-1-1292

```
                            Resolution
                                |
          _____
         /        |          |          |          |         |
        /         |          |          |          |         |
      Entity    Attribute  Logical-    Process    Spatial   Temporal
    resolution  resolution dependency  resolution resolution resolution
                           resolution
```
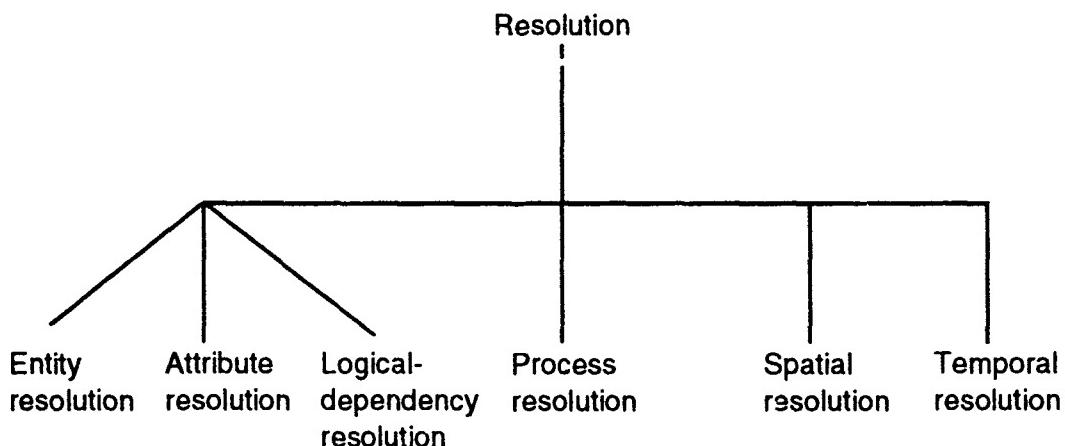
Fig. 1—Aspects of Resolution

---

[1]This study uses "objects" and "entities" interchangeably; similarly, "attributes" and "variables." The "processes" that cause changes in attribute or variable values may be implemented at the level of computer code by "functions" called, in object-oriented programming, "methods."

cumstances, while another may assume that the units always align themselves in a standard formation.

4. Models agreeing in all of the above respects may differ in the resolution of the physical and command-control processes governing changes in entity attributes. We may follow individual battalions, but we may assess attrition at the corps level and assume that the attrition is allocated evenly among battalions on the front line. A closely similar model may instead assess attrition for each individual battalion as a function of its particular battle situation.

5. Finally, the spatial grid and the time step, or the discrete-event equivalent, are also dimensions of resolution.

It may also happen that one model has higher resolution than another in some respects but lower resolution in others.[2] We all know that the relative resolution of two models can be ambiguous, but we often ignore this complexity when talking loosely. In practice, this ambiguity of relative resolution is a common problem. When connecting existing models, we often discover that the allegedly low-resolution model is actually richer in some respects than the high-resolution model. This leads to both technical and sociological problems (as when the organization charged with the high-resolution work has developed a model with some components that are lower in resolution than those of another organization charged with low-resolution modeling).

CONSISTENCY OF PREDICTION

To conclude this section on definitions, let us discuss the important concept of consistency of prediction, which will henceforth be called just *consistency*. Fig. 2 indicates schematically what may be called consistency in the aggregate. Suppose the system being modeled starts (top left) at time T1 in state S(T1). Suppose further that a detailed model exists, which can be represented by a time-generation operator $G(T2;T1)$. The system changes state from time T1 to T2 in a way that can be denoted $S(T2) = G(T2;T1)S(T1)$.

---

[2]Many combat models have inconsistent resolution in that phenomena of interest to the developing organization are treated in more detail than others. So it is that the Air Force often has higher resolution in air-to-air combat than in ground combat, and the Army often treats the air war cursorily, if at all. For some applications such inconsistency is acceptable; for others it is not.
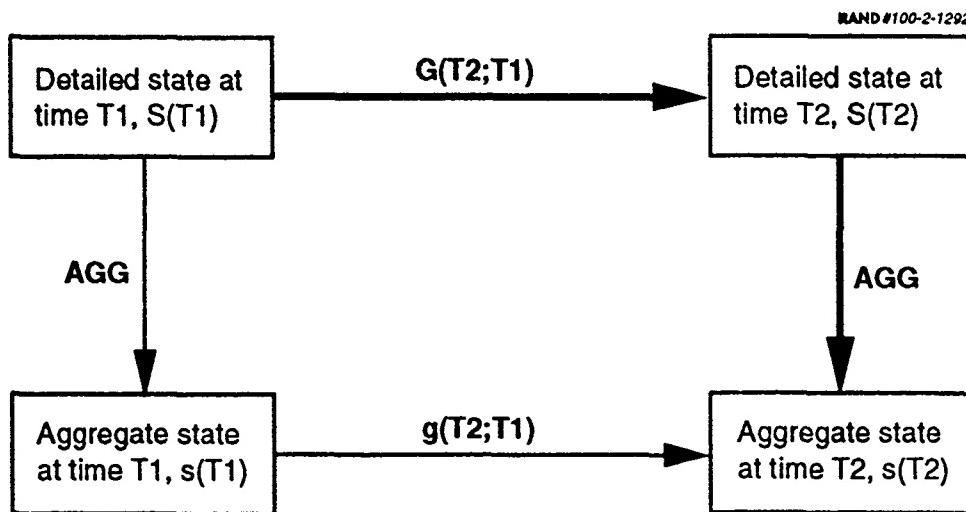
Fig. 2—Consistency in the Aggregate

Suppose, now, that we are interested in a particular aggregation of system characteristics. This might correspond in physics to taking the average over a volume. In combat models it might correspond to aggregating over divisions to get a corps-level depiction of strength. In any case, if we denote the aggregate state by s(T) and the aggregation operator (e.g., one that integrates or adds) by **AGG**, we have (starting from the top left, moving rightward, and then downward):

$$s(T2) = \textbf{AGG } S(T2) = \textbf{AGG } \textbf{G(T2;T1)} S(T1) \ .$$

But suppose instead we had to aggregate the initial state and then used an aggregate model to generate the time behavior of the aggregate system (i.e., suppose we had moved downward and then rightward from the initial state). Would we end up with the same assessment of s(T2)? That is, would we find:

$$\textbf{AGG } \textbf{G(T2;T1)} S(T1) = \textbf{g(T2;T1)} \textbf{AGG} S(T1) \ ?$$

If so, we could say that there is complete consistency in the aggregate.

Fig. 3 suggests a stronger version of consistency. In this diagram we can ask whether the same detailed state at time T2, S(T2), can be generated by both the detailed model and by the process of aggregating, generating the time dependence of the aggregate state, and then disaggregating. The question is:

$$\textbf{G(T2;T1)} S(T1) = \textbf{DISAGG } \textbf{g(T2;T1) } \textbf{AGG } S(T1) \ ?$$

RAND #100-3-1292

```
┌─────────────────────┐      G(T2;T1)      ┌─────────────────────┐
│ Detailed state at   │ ─────────────────▶ │ Detailed state at   │
│ time T1, S(T1)      │                    │ time T2, S(T2)      │
└─────────────────────┘                    └─────────────────────┘
           │                                    │          ▲
           │ AGG                          AGG   │          │ DISAGG
           ▼                                    ▼          │
┌─────────────────────┐      g(T2;T1)      ┌─────────────────────┐
│ Aggregate state     │ ─────────────────▶ │ Aggregate state     │
│ at time T1, s(T1)   │                    │ at time T2, s(T2)   │
└─────────────────────┘                    └─────────────────────┘
```
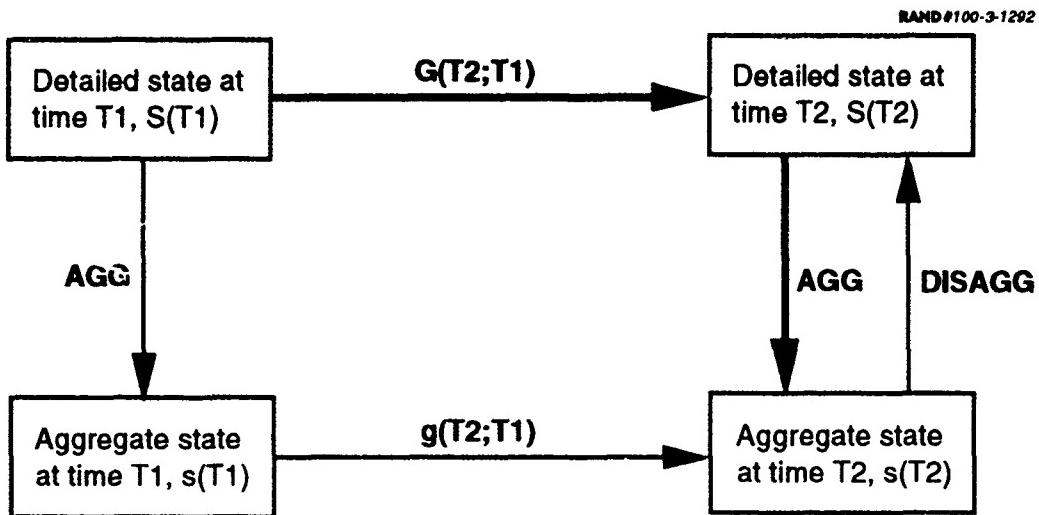
Fig. 3—A Stronger Version of Consistency

If so, we could say that the models are completely consistent. Clearly, this level of consistency would be unusual, because aggregation usually eliminates essential information.[3] There are, however, real-world examples in which we have extra information that permits us to aggregate, generate time behavior, and then disaggregate without losing information. A familiar example from physics is the one of two falling bodies that just happen to be rigidly attached to one another. We can aggregate to look at the center-of-mass characteristics, follow the dynamics of that "effective" object, and then disaggregate to specify where the two physical objects are. An example in the military domain involves army units, which may go through complicated maneuvers while assembling, moving from point A to point B, and then dispersing for combat. So long as it is reasonable to assume that the combat formation is dictated by the circumstances at point B, a combat model could aggregate the forces for movement from point A to point B (i.e., discarding information on their configuration) and then disaggregate at point B before combat begins.

---

[3]One reviewer opined that the most egregious problems of aggregation and disaggregation occur in assessing attrition. He cited an example in which widely different aircraft are aggregated into a pool of aggregated sorties, which then are broken up into sorties for each of the several air-force missions and subjected to aggregation attrition processes. Then the losses are allocated to the various original types of aircraft, with the result that aircraft for close-air-support end up suffering attrition from deep interdiction missions that they would never perform in the real world.

3. WHY VARIABLE RESOLUTION IS IMPORTANT

GENERAL REASONS

Having defined concepts such as consistency across levels of resolution, the next question might be why we would even *want* variable-resolution models (or equivalent families of models). This deserves a more lengthy discussion (e.g., Davis and Huber, 1992), but some of the principal reasons are as follows:

We need low-resolution modeling for:

- Initial cuts (innovation, exploration, etc.)
- Comprehension (seeing the forest rather than the trees)
- Systems analysis and policy analysis
- Decision support
- Adaptability
- Low cost and rapid analysis
- Making use of low-resolution knowledge and data

We need high-resolution modeling for:

- Understanding phenomena
- Representing knowledge
- Simulating reality
- Calibrating or informing lower-resolution models
- Making use of high-resolution knowledge and data

There are subtleties. First, there is confusion between comprehending a model and comprehending phenomena. It is often essential to use high-resolution models to understand phenomena qualitatively (e.g., to discover what the critical factors are), but having obtained such an understanding, it may be desirable to use the simplest model consistent with that understanding to comprehend what we are doing analytically. Even brilliant people often make gross errors when dealing with models having too many variables and relationships.

Another subtlety is that resolution is a *relative* matter. Thus, just as theater-level analysis may require dipping into corps-level analysis selectively, so also corps-level analysis may require dipping into divi-

7

sion-level analysis selectively, and so on. The problem is inherently hierarchical, and one person's high resolution is another's low resolution. At every level, however, there are lower and higher resolution views with the advantages listed above.

Some workers argue that "infinite" computing power is on the horizon and will eliminate the need for lower-resolution models. Others argue that high-resolution models are complex, incomprehensible, and to be avoided. *A more accurate view is that both low- and high-resolution models and analysis will remain critical for the reasons given here, even as computer power continues to increase exponentially.*[1]

SOME EXAMPLES OF HOW VARIABLE RESOLUTION IS USEFUL

There are many current examples of how workers using combat models need to vary resolution. Some worth mentioning are the following:

- Using high resolution to provide a *picture* when the lower-resolution depiction seems too abstract (e.g., to understand when maneuver and counter-maneuver do and do not "cancel out").

- Invoking high resolution for *special processes* within the course of an otherwise low-resolution simulation (e.g., special processes in which physics-level phenomena are critical and cannot be well represented by averages; this arises, for example, when one force has a distinct but situationally dependent qualitative advantage, such as weapon range).

- Using high resolution to *establish bounds* for parametric analyses using lower-resolution models (e.g., bounds on the number of passes per sortie of a fighter-bomber).

- Using high (low) resolution to *calibrate* lower- (higher-) resolution models, recognizing that our knowledge of the world comes at all

---

[1]As noted to me by Dr. Ralph Toms of Lawrence Livermore National Laboratory, this doesn't mean that we need low-resolution models in all applications. Given, for example, well-established high-accuracy computer codes for performing aerodynamic calculations, knowledge of the input parameters, and adequate computer capacity, it is often preferable to use those rather than attempt to apply simpler models through a series of idealizations, boundary-value tricks, and off-line analysis. In most applications of combat modeling, however, the detailed models are not well validated, and the data required for them are highly uncertain (especially for combat conditions). In such applications, we need a combination of low- and high-resolution models for the reasons indicated in the text.

levels of detail[2] (e.g., calibration of killer-victim scoreboards based on a detailed physics-level weapon-on-weapon simulation).

- Using low resolution for *decision support, including rapid analysis of alternative courses of action* (e.g., battlefield decisions under enormous uncertainty, or peacetime acquisition decisions taken years before performance parameters and interrelationships can be fully understood).

- Using low resolution to *generate adaptive scenarios*, as when attempting to provide and maintain context for higher resolution war gaming in which objectives and even tactics depend on higher level considerations and coordination requirements.

Needing variable resolution is one thing, but having it is another. There are many common difficulties. The first is that in attempting to use one model to provide a higher resolution view of the phenomena described in a second model, we often discover that the models simply don't fit together. They may include inconsistent descriptions of the same phenomena, or the descriptions may be consistent only in a way that is quite obscure because of differences of perspective. This is particularly so with higher level analytic models, which necessarily exploit abstractions, because which abstraction is appropriate depends on the application.

Another problem is that when models are allegedly calibrated to each other, they are often calibrated at only a single point alleged to be representative. Little information is typically provided on how quickly the models get out of calibration. And, frankly, a lot of this is done ad hoc. That is, fitting one model to another is often done in off-line analysis that may not even be documented and, even if it were, it would not pass muster in a mathematics class.

A basic motivation for RAND's related project work under DARPA sponsorship is that the issues involved in sound variable-resolution modeling are to a large extent generic, but they are not well known—probably because design continues to be underemphasized in university curricula. With this background, then, let us next consider the

---

[2]Organizations often gravitate toward a mindset in which lower-resolution models are calibrated against higher-resolution models, but not vice versa. This is wrongheaded, unless there is reason to believe the higher-resolution depictions—and the relevant data for them—are more reliable than lower-resolution depictions and data.

types of variable-resolution modeling we might use and then begin moving toward generic concepts, first by working through a very simple example and then by abstracting some principles from that example.

4. TYPES OF VARIABLE-RESOLUTION MODELING

As discussed in some length in Davis and Huber (1992), it is useful to distinguish among three approaches to variable-resolution modeling in the community. These are summarized in Fig. 4.

Selected viewing consists of providing aggregate displays from a more detailed underlying simulation. It is quite valuable and should be encouraged, but it often has problems (e.g., lack of transparency and hidden-variable effects), including an implicit dependence on a mass of high-resolution input data that may be very difficult or impossible for the user to review.

Alternative submodels is the most common form of variable-resolution modeling. By and large, however, the alternative submodels and data bases are inconsistent to varying degrees. While this approach is common and quite useful, it is often inelegant and full of seams. There is, in practice, a wide variation in the quality of variable-resolution models of this type.

RAND #100-4-1292

Common Problems

- Selected viewing
  - —Carry along full resolution
  - —Display lesser resolution as appropriate

  - Quality depends on invisible underlying model and data
  - Hidden variable problems
  - Interactivity requires disaggregation, usually done ad hoc at interface

- Alternative submodels (or model families)
  - —Models have switches
  - —Submodels have different resolution
  - —Submodels may or may not be integrated

  - Inconsistencies
  - Different perspectives
  - Many seams
  - Point-case calibrations, if any

- Integrated hierarchical variable resolution (IHVR)

  - Requires good design
  - Not always possible
  - Choice of hierarchies depends on perspective; limits later choices
  - Can be burdensome

Fig. 4—Classes of Variable Resolution

11

The third approach, *integrated hierarchical variable-resolution modeling* (IHVR), is quite uncommon as a consciously chosen approach, but I have used the method successfully in my own work. By IHVR, I mean modeling that describes critical processes (e.g., attrition, movement, or air attack of ground targets) as being composed hierarchically of subordinate processes. These processes may be fully represented, in which case they generate dynamically the inputs to the higher level processes. Alternatively, they may be replaced, in an approximation, with trivial processes that always provide the same constant inputs to the higher level processes. That is, they may be approximated by parameters.[1]

Importantly, most hierarchical families of models are not integrated and do not have the advantages associated with IHVR design. Instead, most model hierarchies are simply a collection of models of varied resolution along with some procedures for using the more detailed models to calibrate selected input parameters of the lower-resolution models of the hierarchy. This calibration is often painful, inelegant, imprecise, and even dubious. In other cases, the approach is basically sound but painful. Documentation on how the various models of the hierarchies are calibrated against each other is unusual.

We shall work through an example of IHVR in what follows, but Fig. 5 provides a first glimpse at what is involved. This figure has nothing to do with combat modeling; instead, it depicts the relationship among variables in a political model developed for work on deterring opponents in crisis (adapted from Davis and Arquilla, 1991). Each node is a class of variables. When two or more arrows feed into a node from below, it means that the higher level variable is determined by some function of the lower level variables. In a computer version, each node would be a function producing the variable shown (or a set of variables).[2] For simplicity, I have only elaborated the tree of variables for the best-estimate assessment, but there should be similar trees for the other intermediate variables.

---

[1]Alternatively, they may be approximated by very simple processes that generate parameter values that vary with gross situation as defined by a lookup table.

[2]This IHVR approach was used earlier to build large artificial intelligence models of potential Soviet and U.S.leadership reasoning in crisis and conflict (Davis, 1987; Davis, Bankes, and Kahan, 1986). The hierarchical design was critical, because some applications required linking the models to detailed simulation of combat, while others were more profitably accomplished using the higher level aspects of the political models independently, in which case inputs were specified as parameters.

Decision on an
invasion option

Situation*   Best-estimate   Worst-case   Best-case
              assessment of   assessment of assessment of
              option*   option*   option*

Military   Political
prospects*   prospects*

Force   Likelihood   Domestic* Alliance*
balance*   of surprise*

Air   Ground Naval
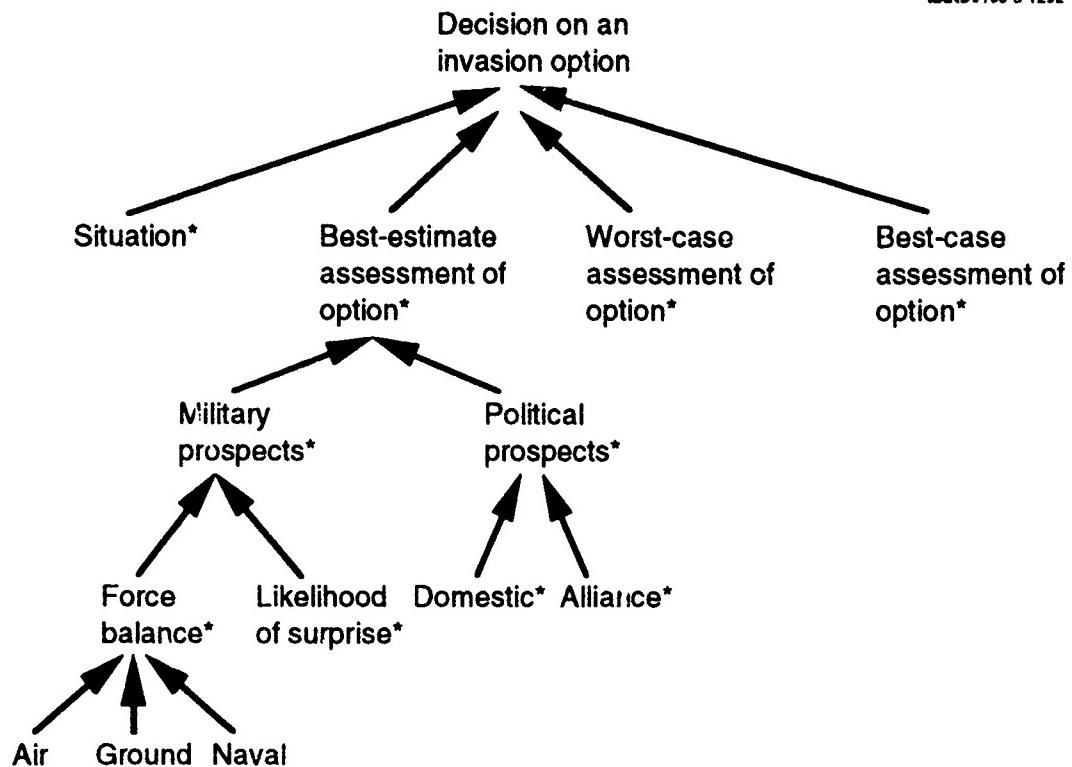
Fig. 5—An Example of Hierarchical Design: Political Decision Models

With this convention, the top-level variable is a decision by a potential invader, who is deciding which of several strategic options to pursue (e.g., no invasion with a reliance instead on bellicose threats, a limited invasion, or a full-scale invasion). This decision depends, for each option, on an assessment of the situation and on best-estimate, worst-case, and best-case (most-optimistic case) assessments of the option in question. These assessments, in turn, may depend on lower level variables such as the likelihood of surprise attack.

Note that the variables are arranged in a perfect hierarchy: each variable affects only variables above it in a single tree. There is no cross-talk between branches and no cycling (i.e., no feedback).[3] This makes it straightforward to implement variable-resolution modeling

---

[3]Feedback can be modeled in this type of structure with a time delay. Thus, the military prospects assessed at the end of one time period may determine the political prospects in the next period.

14

as follows.[4] Wherever we see an asterisk, we can build in an option to generate the variable from lower level variables (higher resolution) *or* to specify the variable directly as a parameter. This model is an example of IHVR. An exceptionally important aspect of IHVR is that we can "see" the relationships among variables in the different levels of resolution. This is a basic element of "seamlessness": we can change levels of resolution without confusion (especially if the variables are appropriately named).

---

[4]I mean here that the *structure* of the problem is straightforward. It may or may not be the case that the more detailed variables change over time in such a way that their effects on higher level functions can be approximated adequately by using average values.

5. A WORKED-OUT EXAMPLE

SIMPLE COMBAT MODELS PERMITTING MATHEMATICAL ANALYSIS

In a preliminary workshop on variable resolution held at RAND in November 1991, participants expressed the need for simple examples that could be worked through in detail and fully understood. With that in mind, let us now consider an exceptionally simple combat-modeling problem. The approach will be as follows: (1) to describe lower- and higher-resolution models developed quasi-independently for the illustrative problem; (2) to examine whether they can be used together as a "hierarchical family of models" by calibrating the lower-resolution model against the higher-resolution model; and (3) upon seeing and describing the difficulties in doing so, to describe how we could instead have used IHVR methods to develop the models in an integrated manner, and why doing so improves clarity and seamlessness. Consistent with what typically happens when we attempt to connect existing real-world models, we will initially proceed knowing only the qualitative model descriptions, rather than including details of the models (the models are defined in detail, however, in Appendix B).

The purpose, then, is first to give a case history of how it is that we so often end up with nonintegrated and confusing models, which we wish were easier to use together so that we could change resolutions at will and, second, to describe a better way to proceed.

The problem we shall consider is simple ground combat between an attacker and a defender engaged in a straightforward head-on-head attrition battle, perhaps at the level of an army attacking a corps. The nature of the low-resolution model is suggested in Fig. 6. In this depiction the forces are characterized in terms of overall attacker and defender "strengths," A and D, measured in equivalent divisions (or something more sophisticated such as the situationally adjusted equivalent-division scores described in Allen, 1992). As detailed in Appendix B, the model assumes that attrition depends solely on the force strengths and attrition coefficients K_a and K_d via the Lanchester square law and the so-called 3 to 1 rule. That is, the model assumes that the rate at which each side loses forces is proportional to the other side's strength. The 3:1 rule states that, at an attacker-to-defender force ratio of 3, the sides will be fighting to a stalemate. The

Defender D	Attacker A
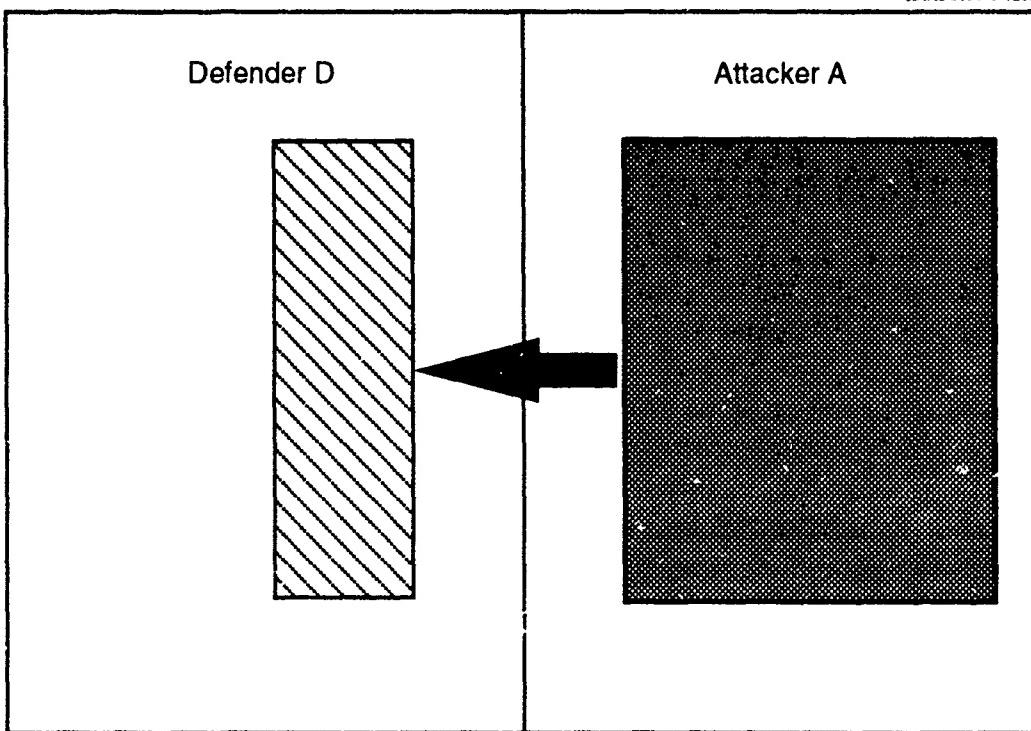
Fig. 6—An Aggregate Model of Ground Combat

argument here is that the defender usually has advantages of pre-
pared positions that provide both concealment and protection. While
this is an extremely simple model, it is one that has been used by
many workers over the years because of its intuitive appeal.

Fig. 7 depicts what, for the sake of this example, can be considered a
"detailed" model—recognizing, of course, that it would be considered
highly aggregate by those working at the weapon-on-weapon level.
(This is discussed in the companion paper by Hillestad, Owen, and
Blumenthal [forthcoming], which goes down to the physics level of
individual tanks shooting at individual tanks.) The "detailed" model
of Fig. 7 breaks the attacker and defender forces down into forces on
the forward line of troops (FLOT, flank forces, and reserves). There is
a military frontage L across which the battle takes place, there is a
specific background of terrain suggested by the shading (e.g., open,
mixed, or rough), and there is some type of defense set up by the de-
fender (e.g., hasty, deliberate, prepared, or fortified), which together
with the sides' tactics determine the "type battle." The model in-
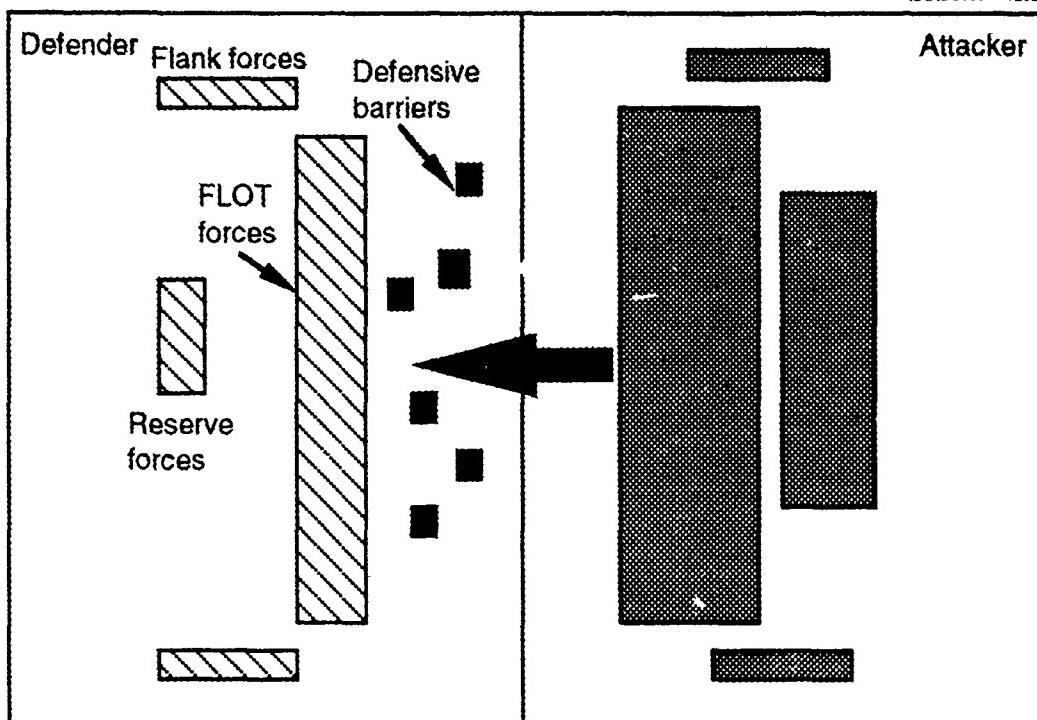
Fig. 7—A "Detailed" Model of Ground Combat

cludes a concept of "breakpoints," under which a side will break off fighting if its attrition is excessive or if it is being severely outflanked. The model requires a great many more parameters than the previous model (see Appendix B). For simplicity, this model also assumes a Lanchester square law and the 3 to 1 rule for forces on the FLOT. That is, the same attrition mathematics applies to this model and the more aggregate one, but the level of detail is different and there may be different parameter values.[1]

---

[1]Even the simple model suggested in Fig. 6, coupled with Lanchester equations in one form or another, are commonly used. See, for example, Epstein (1990). We can also interpret the work of T. N. Dupuy in terms of Lanchester square models (see Dupuy, 1987). The "detailed" model suggested in Fig. 7 is similar in many respects to those used to calculate daily attrition in a variety of theater-level combat simulations, including the RSAS (Bennett, Jones, Bullock, and Davis, 1988, and Allen, 1992). The RSAS, however, uses score-based methods for only short periods of time and uses coefficients that are highly dependent on the operational situation faced by the forces, which changes from time period to time period. The algorithms used are also more complex than the Lanchester square law.

ATTEMPTING TO CREATE A HIERARCHICAL FAMILY FROM EXISTING MODELS

Principle Issues

Given the two models, then, suppose we declare that they constitute a hierarchical family. That is, we observe that one is more detailed than the other, and we assert that the lower-resolution model can be calibrated by using the higher-resolution model. Is this true? Can we make them consistent and move from one to the other at will depending on the resolution we seek?[2]

The usual approach when this issue arises seems to be to run the models, compare the results, find a tuning parameter in the aggregate model, and adjust that parameter until results agree. The circumstances and details of that calibration procedure may or may not be recorded well, and it may or may not make sense to tune the parameter in this way.

Here let us be a bit more careful. Let us assume that the detailed model is correct and see what is required to make the aggregate model reasonably consistent with it. Thus, the issue becomes one of calibrating, approximately, the attrition parameters K_a and K_d of the aggregate model. We do this by calculating the aggregate behavior of the system (the battle between attacker and defender) using the detailed model, and then setting the aggregate attrition parameters so as to obtain roughly the same behavior with the aggregate model.[3]

Generating the Relevant Aggregate Behavior with the Detailed Model

As shown in Appendix B, the aggregate model's parameters K_a and K_d can be related to the dynamics of the battle as follows:

$$\frac{K_d}{K_a} = \text{RLR } F^2$$

$$K_a = -\text{DLR} / F ,$$

---

[2]The whole approach may also be dubious in that the detailed model may not be a reliable basis for calibration when we consider uncertainties in its input parameters and algorithms, but that is another matter.

[3]In the terms of Fig. 2, the detailed model corresponds to G and the aggregated model corresponds to g. We are attempting to see whether the aggregated state s can be more or less correctly generated over time by the aggregated model.

where F is the attacker to defender force ratio, DLR is the defender loss rate (the fraction of the defender's strength lost in a day's battle), and RLR is the ratio of the attacker's and defender's loss rates. That is, if the attacker loses a fraction ALR of its strength per day and the defender loses a fraction DLR of its strength per day, then RLR = ALR/DLR. The time histories of F, DLR, and RLR (and also ALR) can be generated by running the detailed model. As discussed later and as can be appreciated from Appendix B, however, that may not be so straightforward because the variables of one or both models are not always what they seem to be, and it is very easy to make serious errors of calibration. Nonetheless, let us assume that these problems have been avoided.

Calibrating Across Cases and Considering Analytic Context

If the aggregate model were exactly consistent with the detailed model, then K_d and K_a would be constants, but more generally the equations here can be valid only if we allow K_d and K_a to be time dependent. Since we want to make the aggregate model work as well as possible with *constant* attrition coefficients, we must now find good "average" values for $K_a(t)$ and $K_d(t)$ by conducting experiments with the detailed model. But what does this mean?

How do we compute averages? The most obvious and common way appears to be to consider a "representative case" (i.e., a particular scenario), including a particular set of all the input values that seems to be "typical." We shall use that method later, but note first that a more proper way to find a good average would be to consider a *range* of cases, or scenarios, and to develop some kind of weighted sum. We should also consider at what points in time we want the calibration to be best. It follows that we might use an expression such as:

$$\frac{K_d}{K_a} = \sum_{cases,i} W_i \frac{\int_0^{T_{max}} w(s)RLR(s)F^2(s)ds}{\int_0^{T_{max}} w(s')ds'}$$

$$K_a = -\sum_{cases,i} W_i \frac{\int_0^{T_{max}} w(s)\frac{DLR(s)}{F(s)}ds}{\int_0^{T_{max}} w(s')ds'}$$

where W_i denotes the weight given to case i (e.g., "scenario i") and $w(s)$ is a weighting factor for time. The time-weighting would be significant if we were going to use the aggregate model in a context such that we needed it to be more accurate in some time periods than others (e.g., the first few days of a battle). Similarly, we might be interested only in times less than some T_{max}.[4]

In principle, we might set up a formal criterion such as choosing values of K_a and K_d that minimize the expected discrepancy between the consequences in a larger application of using the detailed and aggregate attrition equations for a specified set of cases and specified time intervals. This would then *imply* appropriate weighting functions. It should be evident, however, that attempting to do this type of thing is very complex and depends sensitively on application details. In practice, we must usually be satisfied with less than optimal choices of the coefficients, choices defined by relatively simple averages such as those shown, and using simple weighting factors (e.g., treat all cases within a particular set as equally important and ignore the others; similarly, treat all times as equally important for times less than T_{max}).

Even if we are able to define and take such averages, it may or may not make sense to do so. Do we have reason to believe that the distribution of values for the coefficients K_a and K_d will be "normal" around some natural average? Perhaps instead the distribution is multimodal (see, e.g., Hillestad, Owen, and Blumenthal, 1992), in which case no single calibration for an "average" case may be useful.[5] Further, even if there is a natural average to be taken, what is it? What weighting factors should be applied for the cases and for different times? Should they be based on likelihood, importance, or both? Importance to what? How long should the averaging interval T_{max} be? The answer is that the "right" weighting and interval depend on the context in which the results of the modeled combat are to be used (e.g., upon force levels, frontages, etc., and, indirectly, on political-mil-

---

[4]It is common in simulation models to readjust many "constants" at the beginning of each time step or when certain major events occur such as a change in the nature of battle or the arrival of reinforcements. As a result, dynamical equations such as Lanchester expressions often need only be reasonable approximations for a relatively short period of time. They should be calibrated with this in mind.

[5]As an example here, suppose that in some battles the defender was out in the open and in other battles he was behind fortifications. "On average," across battles, the defender might barely hold his own, but in particular battles he might be decimated or might decimate the attacker. The "defender advantage," then (which name we give to K_d/K_a because of its effects on ratio of loss rates), is not well described by a single value.

itary scenario, strategies, and other variables). Regrettably, such important issues are seldom discussed when claims are made about one model having been calibrated against another.

Working Through a Representative Case

Let us now work through what might be called a "representative case" for corps-level battle. We will not bother with averages over cases. Instead, we will just do the calibration for this case, a dubious but standard practice. For simplicity, we also assume a simple treatment of reserves in which FLOT forces remain on the FLOT and reserve forces reinforce them subject to various constraints discussed later. We will also ignore the issue of flanks. There is no concept of units. Our example will assume 9 equivalent divisions (EDs) attacking 3 EDs across a 50 km frontage, with no flank forces and no breakpoints. An "equivalent division" is a division that is as effective as a standard armored division, even though its composition may be significantly different.

Figure 8 shows the "actual" behavior of $K_d(t)/K_a(t)$, which can be called the aggregate "defender advantage," and compares it with what would apply if behavior in the aggregate model were correct. That is, the curve marked "actual" is the result of using the detailed model to find the time-dependent ratio of the aggregate model's defender advantage using the equation above. If the aggregate model were exact, the ratio $K_d(t)/K_a(t)$ would be constant at a value of 9 (see Appendix B). In fact, it is not constant, but averaging the "actual behavior" over the time interval shown produces an average value of 8.4, with a small standard deviation of only 1.1. The calibration of K_a yields a value of 0.026.

How well does the calibration work? Figure 9 shows, for the same case of 9 EDs vs. 3 EDs, the time dependence of attacker and defender forces as predicted by the detailed and aggregate models. The agreement is excellent. Aggregation appears justified, at least as a good approximation.

Sensitivity Testing to See If the Calibration Holds

Now, however, let us try to use the model for a different case, one involving 25 EDs attacking 5 EDs, still on a 50 km frontage and with nothing but force levels changed. Figure 10 shows the results. Here the agreement, using the same calibration as before, is miserable. If
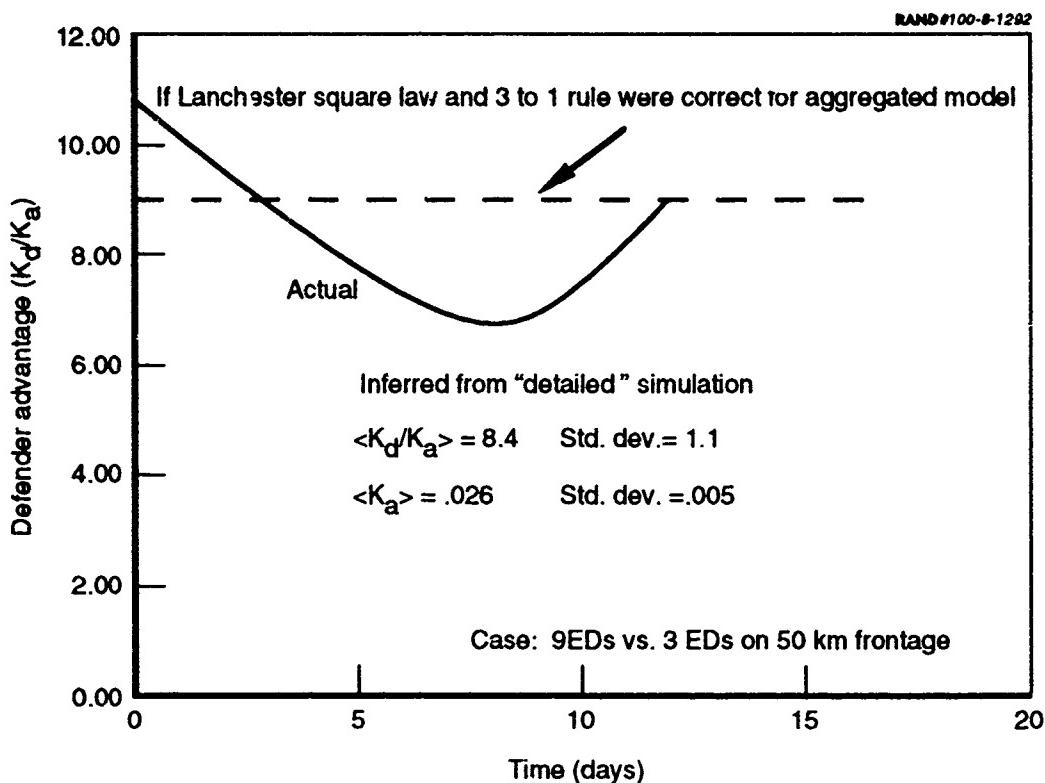
Fig. 8—A First Calibration of the Aggregate Model

we look now at the aggregate defender advantage K_d/K_a over time for this new case (Fig. 11), we see that the value is even less nearly constant than for the previous case, and nothing like the constant value of 9 that would apply if the aggregate model were exact. Obviously, the "representative case" wasn't all that representative. Or, to put it otherwise, the calibration was not robust. When dealing with more complex models, especially when they are being treated as black boxes, people are often surprised to find that calibrations don't work very well—i.e., that they prove not to be robust. Further, this may be discovered only after a long time, because there may have been little initial sensitivity testing after the initial calibration based on a "representative case." Initial sensitivity testing is especially difficult with large and complex models.

Diagnosing the Breakdown of Calibration

Why are we getting this behavior? If we go back and compare the original physical pictures (Figs. 6 and 7), we may guess the answer. When dealing with more complex models, however, the reason for the

RAND #100-9-1292
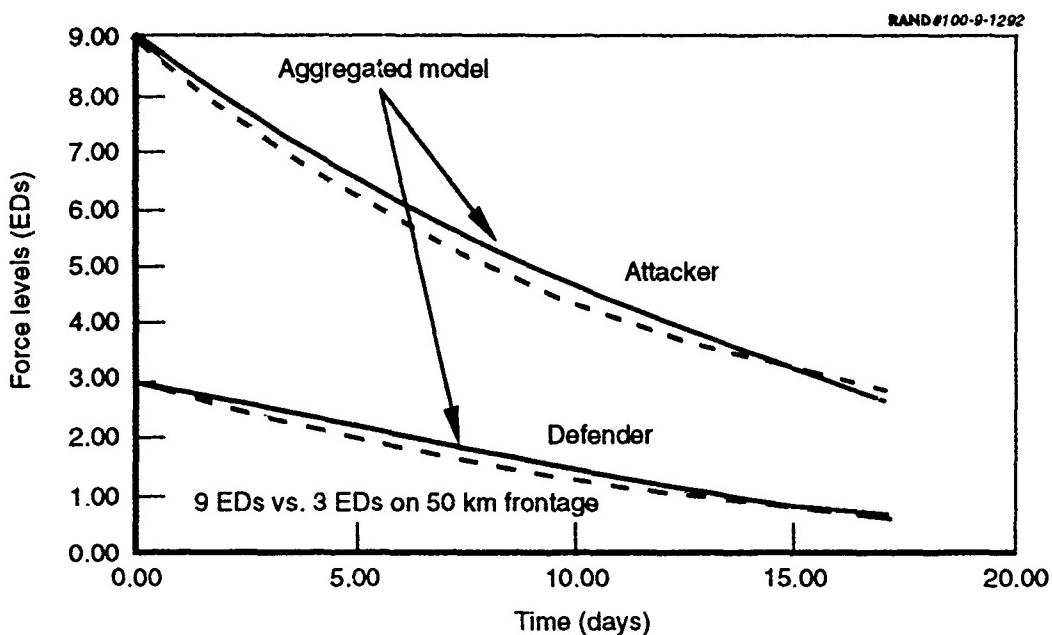
Fig. 9—Consistency (in the Aggregate) over Time

RAND #100-10-1292
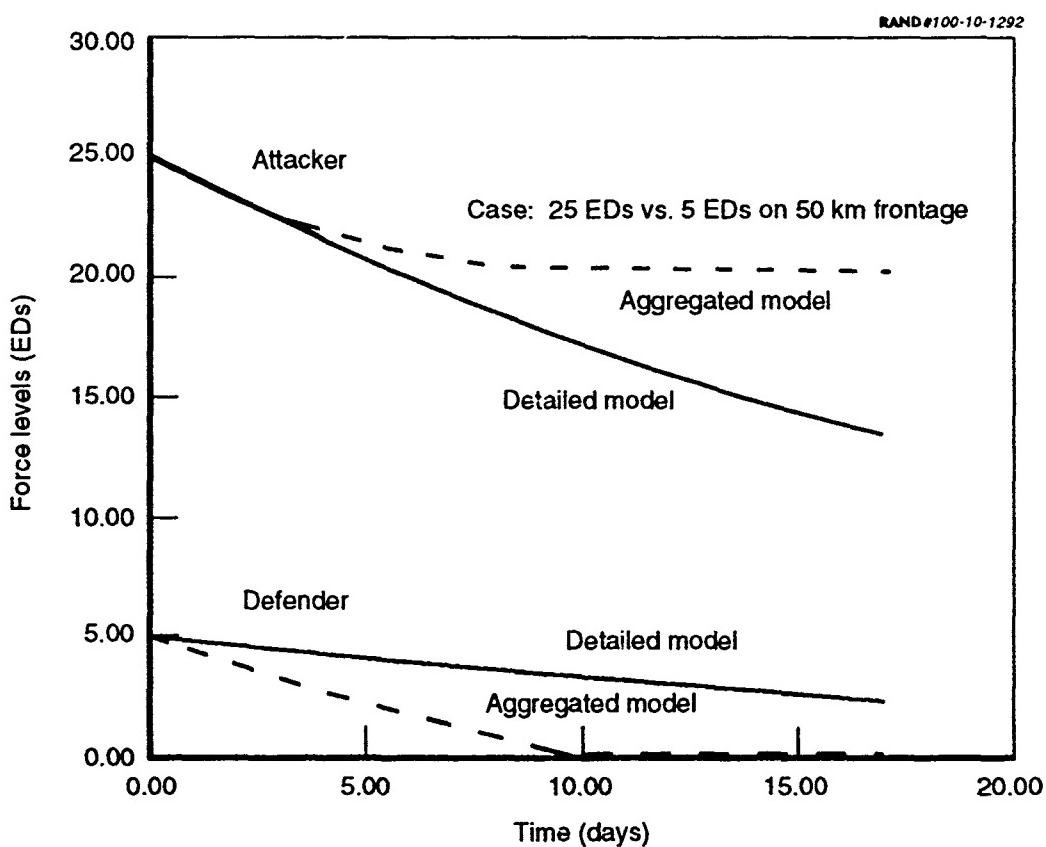
Fig. 10—Failure of Calibration for Other Cases

RAND#100-11-1292

$<K_{dtot}/K_{atot}> = 21$; $<K_a> = 0.008$

Actual

If Lanchester square law and 3 to 1 rule were correct for aggregate model

25 EDs vs. 5 EDs on 50 km frontage

Defender advantage (K_d/K_a)

Time (days)
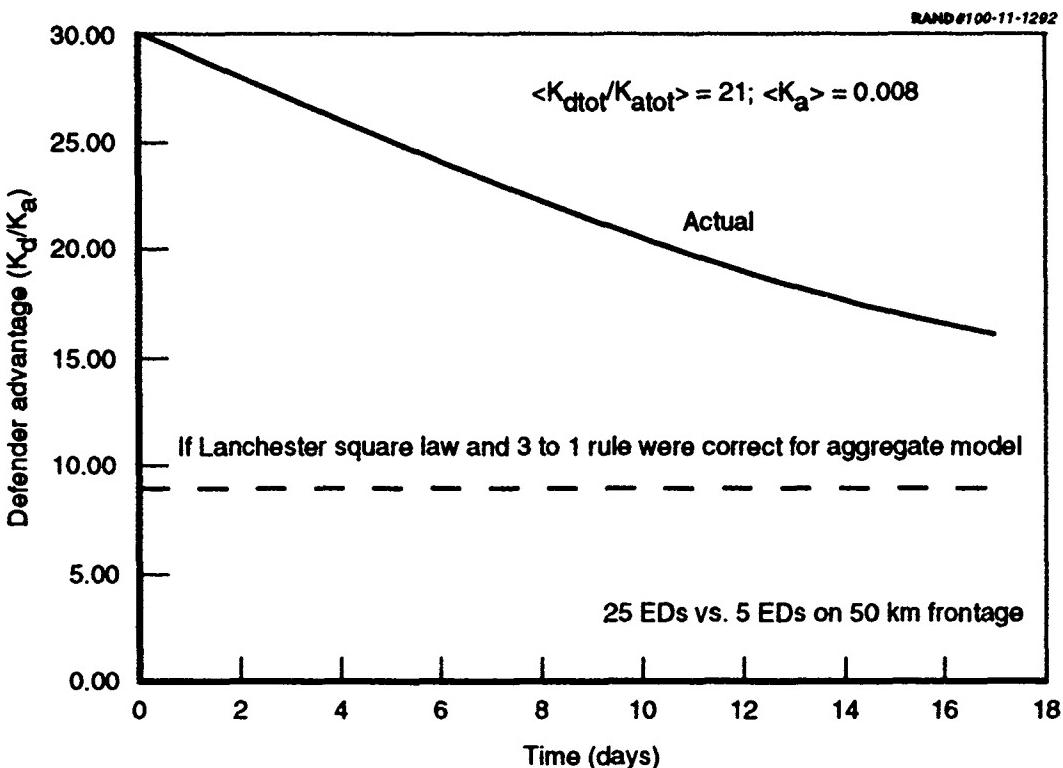
Fig. 11—Badly Nonconstant "Constants"

breakdown of calibration might not be obvious, especially to those attempting to treat the models as "black boxes." The reason in this case can be understood from Figs. 12 and 13, which indicate how the fraction of the attacker's and defender's forces on the FLOT vary with force levels.[6] The detailed model includes the concept of "shoulder-space limits," which states that the attacker will try to squeeze as much on line as possible up to the point at which an equivalent division has less than a minimum frontage.[7] Thus, as we increase force levels, the fraction on line goes from 1 to something smaller (Fig. 12). This particular figure comes from making particular assumptions

---

[6]For the sake of the example, it might have been better to show these fractions as a function of time, because workers often seek to diagnose problems by looking at model behaviors over time rather than understanding the causes of that behavior. Figs. 10 and 11 are more useful, however, in understanding the underlying phenomena.

[7]The concept of a shoulder-space limit is difficult to understand without detailed analysis that accounts for the mutual interference of maneuver vehicles that are too closely spaced and the increased vulnerability of forces that are too concentrated. In World War II, divisional frontages were sometimes as small as 1 or 2 km, but in modern warfare, doctrine typically calls for frontages of 10 km or more, even for attackers.
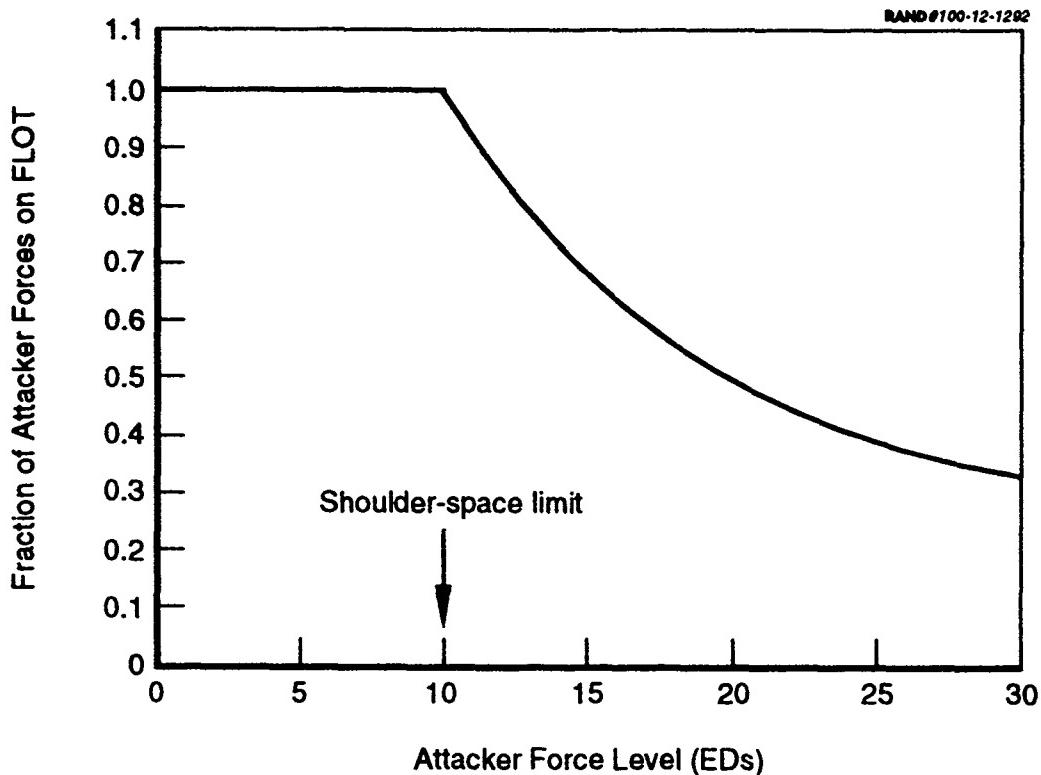
Fig. 12—Attacker May Be Shoulder-Space Constrained
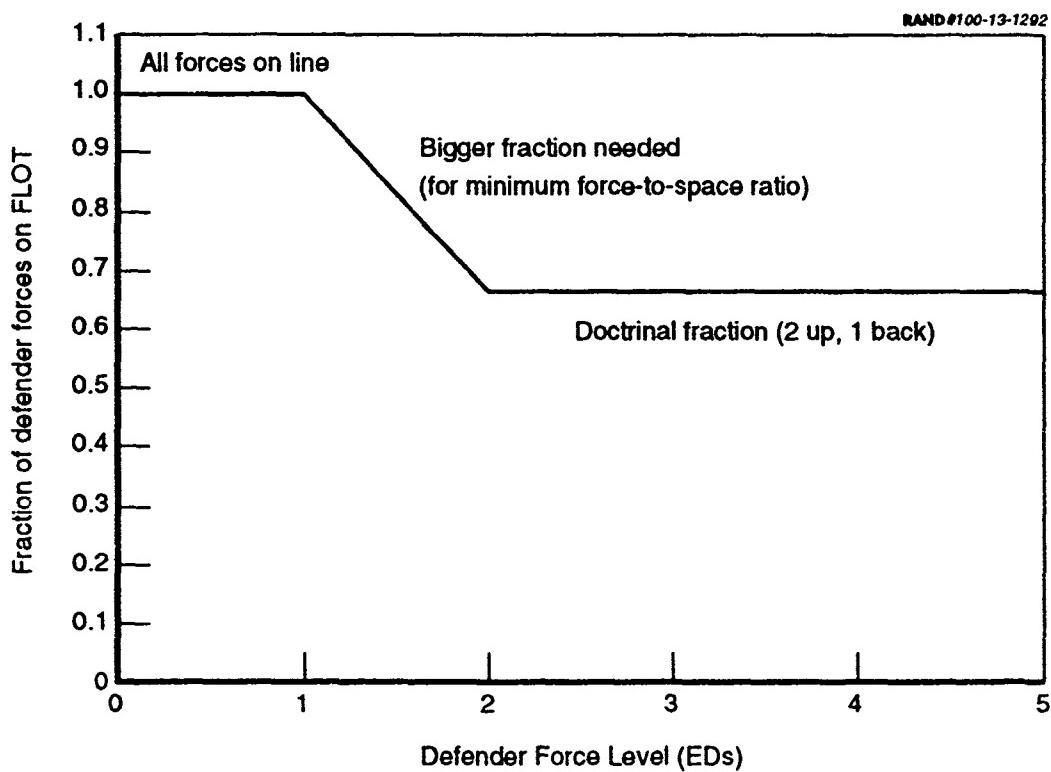
Fig. 13—Defender May Not Be Able to Keep Reserves

about the shoulder-space limit (about 10 km /ED), and on details of the tactical decision rules, but the form is what matters to the concept.

The story for the defender is a bit different (Fig. 13). The detailed model assumes that the defender tries to keep a doctrinal fraction forward (here shown as 2/3, corresponding to the "two up and one back" rule) but never wants the force-to-space ratio to fall below a certain threshold. Thus, if the defender's overall force levels shrink too much, he must put an increasing fraction of his forces on line.

These considerations can lead to a complex time dependence of the defender advantage. In some cases (e.g., 15 EDs vs. 3 EDs), the aggregate defender advantage, K_d/K_a, actually oscillates as a function of time (Fig. 14)! First it is high because the defender has a larger fraction of his forces on line; then, as the attacker suffers attrition, he is able to put a larger fraction on line, reducing K_d/K_a. Thereafter, the defender finds himself having to keep a smaller fraction of his forces in reserves, which pushes the ratio up, and so on.

PATCHING THE AGGREGATE MODEL

Someone who distrusts aggregation generally would stop even experimenting with aggregation at this point (if he even went this far). However, a believer in aggregation, a systems analyst with faith in reductionism, would plunge on. He would say, figuratively,

> Of course. I was stupid. But now I can fix the problem. What's going on is that there is an interaction between frontage and force levels. The aggregation works well only if there is a constant fraction of forces on line over time. But that can't be true if I consider Thermopylae as well as more typical circumstances. In mountains, where the frontage is small, there will be a much smaller number of forces on line. And in open fields, there may be more forces on line (for the attacker) than I assumed previously. So perhaps I need to distinguish three cases: open, mixed, and mountainous terrain; and also three levels of forces: tiny, average, and big. Let's assume that the standard case is mixed terrain and average force levels. We can now develop a slightly more complex model, which we will need to calibrate for the various cases. This, however, should "do it." We still don't need to muck around with distinguishing between FLOT and reserves or to worry about the detailed frontage, etc.

Although not shown here, such an approach can be applied with the result that the model works not only for the original "representative case" but also for the other cases looked at previously. The believer in aggregation is pleased. Instead of working with constant coefficients
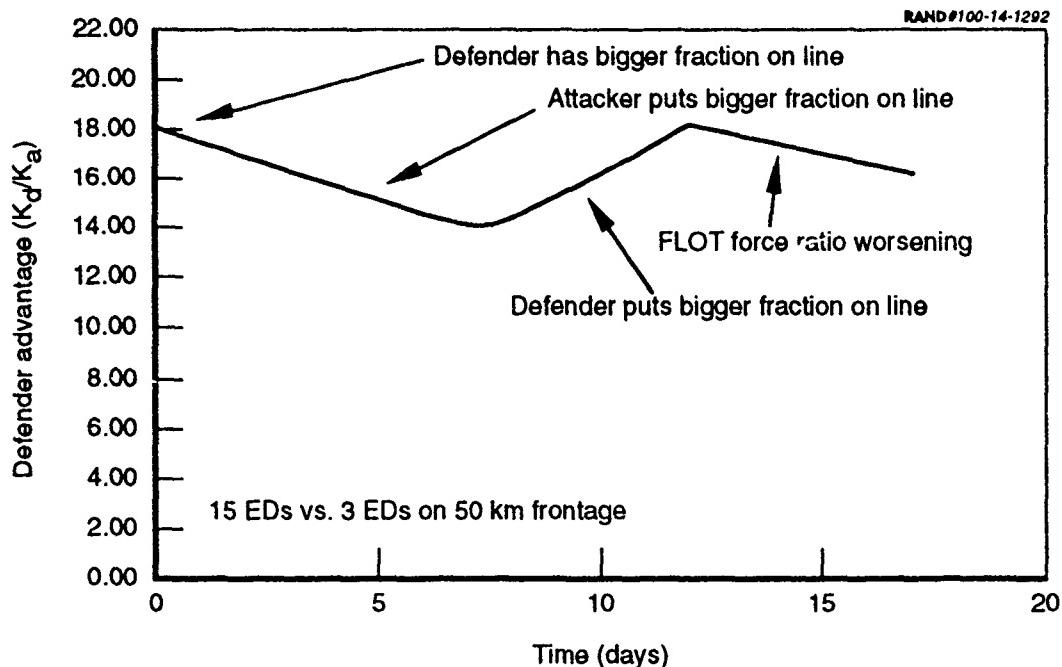
Fig. 14 —A Case of an Oscillating Defender Advantage

K_d and K_a, however, he now has a table-lookup system in which these coefficients vary as a function of the type terrain and qualitative force size.

One trouble, of course, is that the enriched aggregate model still omits many features of the more detailed model, and when we look at some of the issues sensitive to those features, the aggregate model fails. For example, consider the use of "breakpoints," where a side retires from the field if it loses 30 percent of its strength or if it is faced with serious flank problems. It is easy to use breakpoints in the detailed model but not in the aggregate model. Further, if we tried to build a breakpoint into the aggregate model, we would have to go through yet another calibration process, because the breakpoint in the aggregate is not the breakpoint for engaged forces. We could handle this analytically with a bit of effort, but with more general expressions than Lanchester equations, the enrichment would be complex.

There are any number of other detailed issues that could be reflected in aggregate models with proper calibration, but the point is that if we try to address them one-by-one, as "patches" to an originally simple model, the result becomes more and more burdensome and the calibrations more and more difficult to define and track. That, how-

ever, is the most common history suffered by aggregate models: they are found wanting, are patched, then patched again, and eventually become tedious and complex.

RECAPITULATION

Let us now recall the generic issues with respect to calibrating aggregate models to detailed models. Our first approach to the aggregate model was to replace the sum over cases by a single, hopefully representative, case. That represented a leap of faith. The second version of the model amounted to saying that the behavior over cases is complex, but breaks down into a few classes: open, mountainous, and mixed terrain; and tiny, average, and big forces. Within each of those, we could replace the sum by a representative case of that class.

We were also assuming, implicitly, that the appropriate time average was a simple one weighting all times equally over the duration of a representative battle. That would be reasonable enough unless the attrition estimates were embedded in a larger model in which some times are more crucial than others.[8] Even here, however, we should ask "But how long is the representative battle over which the averages should be taken?"

The claim of this example is that we have walked through a relatively common development history for high- and low-resolution models. We end up with a "hierarchy" of two models, so we have variable resolution, but the relationships between them are sloppy and neither integrated nor seamless. The aggregate model was not really designed to be consistent with the detailed model. Instead, its form was postulated, found wanting, and then patched. In more typical cases, it would be patched again and again. Further, the initial effort to calibrate the models was simplistic and ill conceived. Could we do better? That is the subject of the next section.

---

[8]As an example, in strategic mobility modeling, it is not unusual to include an explicit time-weighting because we want to make lift decisions that minimize shortfalls in delivered men and equipment, but it is more important to do well initially than later, after substantial forces are in theater.

6. INTEGRATED VARIABLE-RESOLUTION HIERARCHICAL MODELING

Suppose now that we take a different approach to the same problem. If we started over to redesign the models jointly—to develop an integrated family of two models by looking at the internals of the models rather than their black-box behavior—we would quickly find ourselves confused by differences of nomenclature and concept. For example, looking at the models in Appendix B, we discover that the models use the same notation for very different things, just as two more realistic models developed separately would. In particular, A and D signify the total attacker and defender strengths in the first model and the on-FLOT strengths in the detailed model. Similarly, K_a and K_d are obviously not the same in the two models, even though the notation seemed natural to both models when they were built.

These problems of confusing names may seem straightforward to discover and deal with, but there are more subtle ones as well. For example, the aggregate model has a different *concept* than the detailed model of how to represent terrain; it's not merely a matter of aggregation but also a matter of *perspective* or approach. The simplest version ignored terrain altogether, but the patched-up version has a concept of type terrain, with values of open, mixed, and mountainous. The detailed model also has a concept called type terrain, with roughly the same values (open, mixed, and rough), but it is a very different concept, one applying on a different scale of spa⋯l resolution. The detailed model assumes that combat only occurs on portions of the geographic frontage that are suitable for armored operations. It refers to a "military frontage" L, which is only a fraction of the geographic frontage. The detailed model's notion of terrain type is one that applies for that militarily usable frontage. Thus, we might use the detailed model for analysis in mountains, assume a small militarily usable frontage corresponding to narrow valley roads, but consider the terrain on that frontage to be "open" or "mixed." By contrast, someone using the aggregate model might look at the overall region and assert that the type terrain is "mountainous." The potential for confusion here is enormous.

This example is not even especially contrived. A few years ago RAND conducted a study concerned with estimating the so-called "operational minimum," the minimum force level with which NATO could reliably defend Western Europe even if Pact forces were at parity

with NATO. (Despite common views to the contrary, it turned out [Davis, 1990] that no such theoretical minimum exists.) In the course of studying why people were coming up with varying claims, we discovered that major organizations throughout NATO used vastly different methods for characterizing and dealing with terrain, and the discrepancies were generally either unnoticed or ignored, even though they had big effects.

The first step in developing an integrated approach, then, is developing a complete data dictionary with a consistent and intelligible notation. This is not an easy process in general. An important concept here is the notion of a *reference model* that contains all the variables of either of the two original models, plus additional variables needed to clarify relationships and complete the physical picture. These variables, however, must have sensible names that clarify rather than obfuscate. These names will often be different from those of one or both of the original models. In an abbreviated form (i.e., giving only short-form definitions for a subset of the variables), the results might look as shown in Fig. 15 (see also Appendix B).

RAND #100-15-1292

A, D	Attacker and defender force levels
A_{flot}, D_{flot}	Attacker and defender force levels on the FLOT
K_a, K_d	Coefficients in the attrition calculation for FLOT forces
K_{atot}, K_{dtot}	Coefficients in the aggregate attrition model (for total forces)
L_g, L	Geographic and militarily "usable" frontage
DDF_{min}, DDF_{max}	Minimum and maximum defender divisional frontages
ADF_{min}, ADF_{max}	Minimum and maximum attacker divisional frontages
terr, type battle	Correction-factor parameters adjusting effective strengths to account for terrain and circumstances of battle (e.g., defender's preparations)
A_{break}, D_{break}	Force levels (fraction of original) at which attacker and defender armies break off battle ("breakpoints")

Fig. 15—Defining Variables Consistently in a Reference Model

To avoid confusion, the original models migh be recoded in terms of the reference model's variable names (and, also, the names of processes). Sometimes, indeed often, this is considered unnecessary or too expensive, but there is a price paid in comprehensibility, maintainability, and the potential for errors. A compromise, of course, is to build an interface that contains the mappings. Then we can think in terms of the reference model variables even though the models continue to operate with their original notation. This may be an adequate approach if indeed the only issues are names. In practice, however, there may also be a need to change algorithms, in which case the confusion problem can be severe.

Given the reference model, I recommend as a next step literally drawing pictures showing functional relationships. Figure 16 shows such a picture for the original models (simplified to ignore flank and breakpoint issues, but with the "patches" included for the aggregate model). We can think of these figures as being the skeletons of data-flov diagrams in which the variable names are place holders for bubbles such as "Compute DLR, ALR, and RLR." We see in this depiction of the original models that the variables were confusing, non-hierarchical, and different in perspective. For example, the detailed model high-
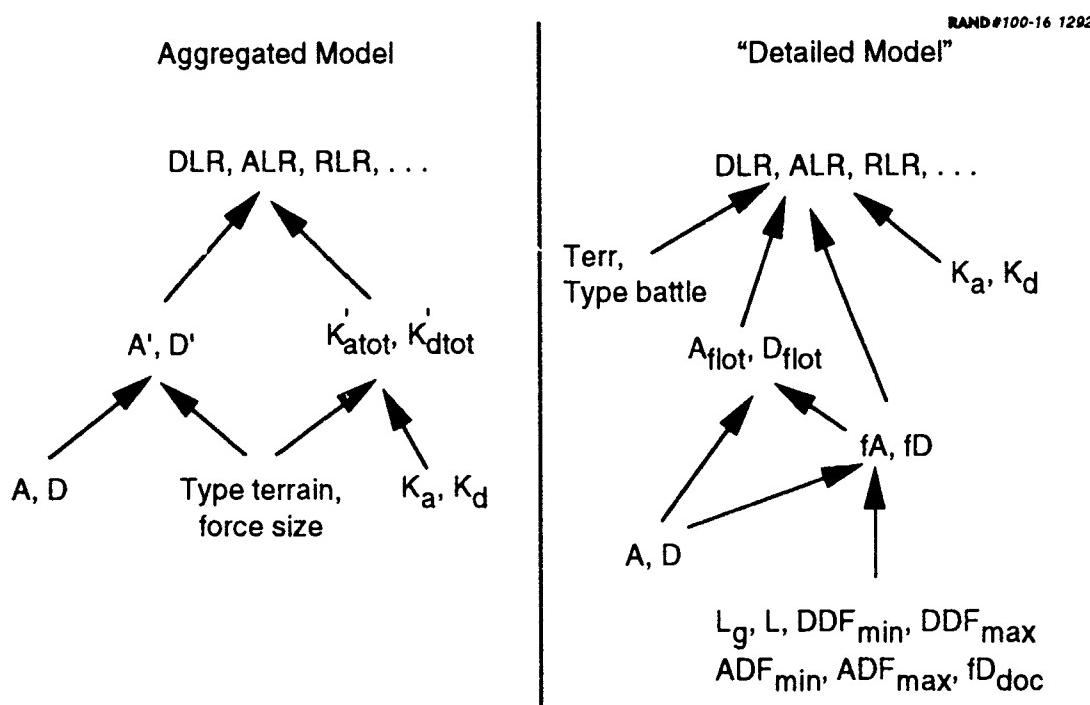
Fig. 16—Original Relationships with Renamed Variables

lighted the concept of the fraction of forces on line (see the variables A_{flot} and D_{flot}). That concept obviously doesn't appear in the aggregate model. However, the aggregate model has a concept the detailed model did not: it calculates "effective" force strengths (denoted by primes), which are obtained by multiplying nominal force strengths by correction factors for terrain and force size. By contrast, the detailed model treated terrain in a different way and never used "effective force size" as a concept. Which is right?

If we examine the models with this confusion in mind (and here we need to be looking directly at the computer code unless the model is fully specified outside the program), it becomes clear that we can unify the descriptions as suggested schematically in Fig. 17: Some of the differences between the two models (e.g., the ones mentioned previously) were arbitrary and can be eliminated.

Note that we now have hierarchical structures. Further, most of the concepts of the high- and low-resolution models are the same. And,
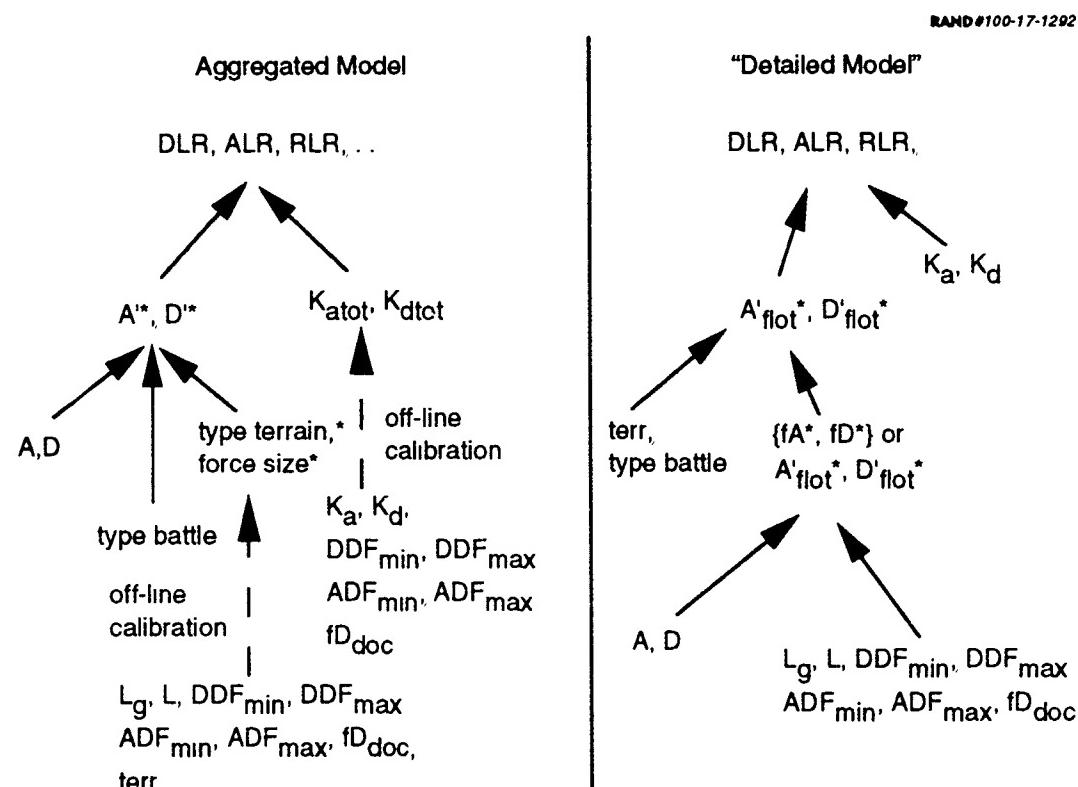

Fig. 17—A Revised and Hierarchical Design with Alternative Resolution and Perspectives

finally, the flow for calibrations is explicitly indicated. In most cases the algorithms for those calibrations are straightforward because the concepts *do* correspond. Where the concepts are not the same (as in the treatment of terrain), we must either make them consistent by changing one model rather fundamentally or develop the appropriate calibrating algorithms. Thus, if we wish to retain the concepts of type terrain and force size in the aggregate model, we need to specify how we can infer their values from the variables of the detailed model. The algorithm for doing so may or may not be satisfying; it will usually be heuristic.

Looking at the detailed model on the right, we see a variable-resolution hierarchical design with several natural levels. Again, asterisks indicate levels at which - can either provide inputs directly (by specifying the asterisked variables as parameters) or generate the information by executing more detailed functions dependent on the variables shown by arrows coming up from below. Assuming we programmed in the appropriate switches, we could choose to run the detailed model at maximum resolution, or at a level at which we specify the forces on the FLOT directly, which might be useful. Alternatively, we could use the aggregate model at two levels of resolution (i.e., in addition to having a variable-resolution model, on the right we could have an alternative aggregate model with a slightly different perspective than the lowest-resolution version of the variable-resolution model). The aggregate diagram also shows that parameter values of type terrain and force size could be calibrated off-line in terms of the detailed model's parameters. This, of course, would require appropriate averaging over cases, as discussed earlier.

The family of models indicated schematically here are the same substantively as before, but they are now integrated: Relationships are explicit, notation is consistent, and it is clearer how one set of concepts and variables flows into another. Some of this is notation, some of it is design choice, and some of it is the picture itself. *The claim here is that this revised design is more "seamless" because of the integration.* For example, suppose we had been using the detailed model and decided to switch to the aggregate model. A mere glance at Fig. 17 (and Fig. 15) would indicate two things. In the new model we would be thinking in terms of modified coefficients K_{atot} and K_{dtot}, which can be calibrated to parameters of the detailed model but are not identical with K_a and K_d. We would also think exclusively in terms of total forces, not FLOT forces, but the effective strength of the total forces would indirectly account for the variability of force fraction on the FLOT by applying corrections called type terrain and force size, which can be calibrated to concepts of the detailed model. This

seems conceptually straightforward and does not require any mental lurches.

Although this approach to model design is uncommon, it often works and works well (e.g., Davis, 1987). Figure 18 depicts a model recently developed in Germany by Reiner Huber (see Davis and Huber, 1992). The top level of this model describes an overall effective "force ratio" for the theater. That, in turn, is computed by combining measures of ground force potential and air-support potential, which in turn depend on more detailed variables, down to the level of sortie rates, kill probabilities, and terrain characteristics. The motivation for this model (called GEFRAM) was the need for policy makers and general officers to be able to understand certain military balances without going through complex simulations. The important point in this study is that the model's architecture can be understood at a glance, and there is true variable-resolution capability from physics level relationships such as range-payload data up to a corps-level force-ratio calculation—all within a single model.

RAND#100-18-1292
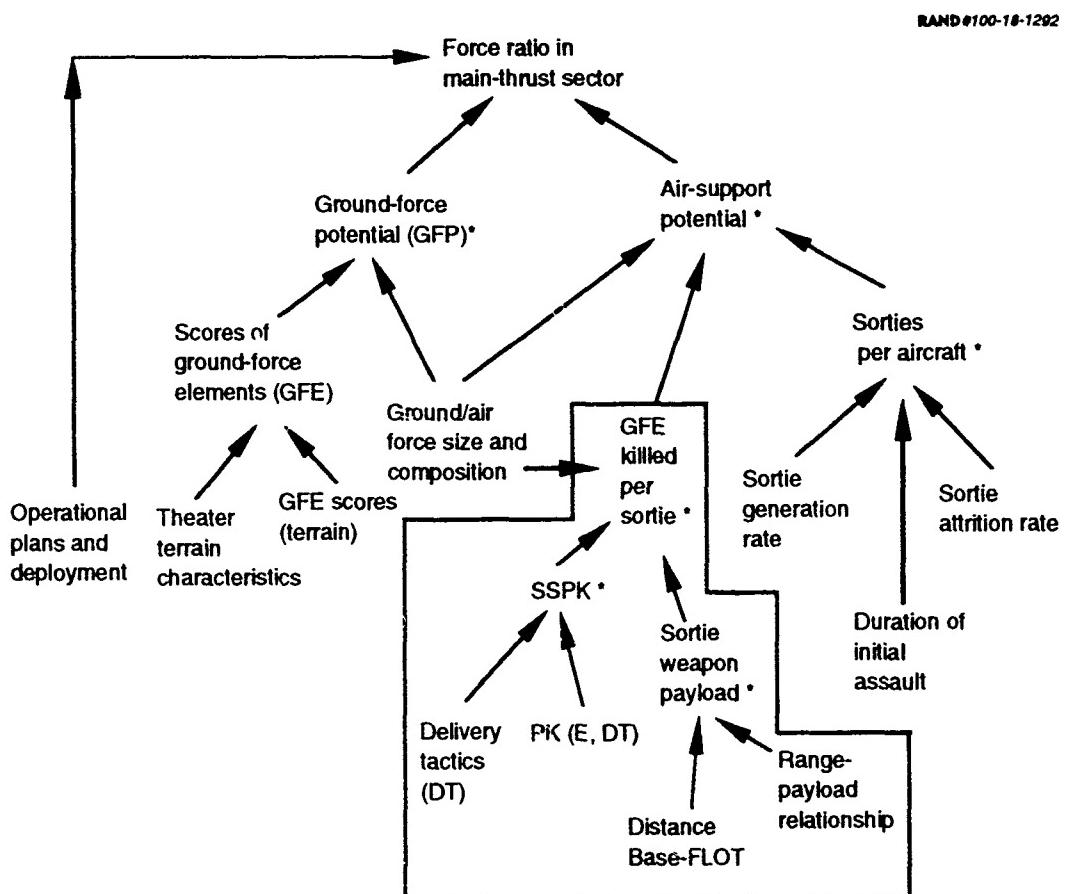
Fig. 18—Another Example of IHVR Design

7. DISCUSSION AND RECOMMENDATIONS

GENERIC CHALLENGES

Let us now consider some of the more generic challenges of variable-resolution modeling and what kinds of approaches make sense. Figure 19 summarizes such challenges, using a terminology associated with pairs of models, even though in practice we often want multiple levels of resolution. As shown, it is essential to get the concepts and variable names straight and to generate a complete set of variables and functions so that we have a representation of the "entire" system (this is the "reference representation" discussed earlier). It is very useful to draw the relationships and mappings. The next item in the list is to decide on the *form* of reasonable aggregate equations. The last section assumed that the aggregate equations had the same form as the detailed equations, but that is usually not the case. Further, it is usually not a good idea to assume that the aggregate expressions will be simple and intuitive.[1] We need to add a dose of *theory* to inform our hypotheses. That said, experience suggests that intuition and first-order theory are often inadequate, especially when (as in many simulations) problems do not lend themselves to closed-form analysis. Experimentation with higher-resolution models is often necessary in order to develop a good sense of proper aggregations. Even so, a moderate amount of equation shuffling can pay big dividends. The last item in the list of Fig. 19 addresses the problem of defining appropriate averages over relevant cases, averages defined with appropriate weighting factors, which unfortunately will be sensitive to the analytic, educational, or operational context.

ON THE GENERALITY OF INTEGRATED HIERARCHICAL METHODS

In this paper I have emphasized integrated hierarchical variable resolution (IHVR) primarily because it has such a high payoff when it

---

[1]As discussed in Horrigan (1991), standard aggregated models of combat often assume independent events and ignore the role of spatial relationships. The result (configurational errors) can be errors of a factor of 2 or more in the assessed relative goodness of alternative weapon systems. The example in the last section is actually a special case of a configuration problem in that the relationship between FLOT forces and reserves is critical but cannot in general be accounted for by a constant scaling factor. Other more complex examples are given in Hillestad, Owen, and Blumenthal (1992).

- Getting the concepts and n⌐mes straight.

- Completing sets of variables and functions (i.e., defining the reference model).

- Drawing relationships and mappings.

- Deciding form of reasonable aggregate equations relative to detailed equations (requires theoretical analysis).

- Finding conditions under which aggregation equations might be reasonably valid (requires theoretical analysis).

- Expressing aggregate-model parameters in terms of outputs of detailed model (requires theoretical analysis).

- Deciding on cases (e.g., scenarios) to be distinguished and how to make calibrations for each case—e.g., how to determine weighting factors over case and time so that calibrations will be appropriate for context of larger application (requires theoretical analysis).

Fig. 19—Generic Challenges in Variable-Resolution Modeling

works and because it is not well understood in the community. A few observations are appropriate, however. First, note that the hierarchies treated here involve *processes* (e.g., "Compute attrition"), not objects or entities. This is significant because the powerful methods of object-oriented modeling and programming are primarily focused on objects, not processes. While they make hierarchical modeling straightforward, the hierarchies are in another dimension of the problem (e.g., army groups break into armies or corps, which break into divisions, which break into brigades, and so on). While hierarchical representation of objects is rather widely valid and natural in combat modeling, *straightforward* hierarchical modeling of processes is only sometimes feasible. More generally, the relevant processes have a more complex relationship to each other, with connections across branches of the hierarchical tree and, in some cases, with iterations or cycles of data flow. It follows that to exploit IHVR methods it will be necessary to develop appropriate approximations that break these cross-branch interactions and cycles and compensate, for example, by adjusting coefficient values from time to time in the simulation as gross features of the situation change. This will require theoretical analysis explicitly separating the different spatial and temporal

scales that are natural to the problem (e.g., road marches often occur over many hours, while close combat at the brigade level may be completed in tens of minutes).

There are a variety of other complications (see Davis and Huber, 1992), including the tendency to underspecify models. This leaves programmers to fill in details, which they often do in ways that limit flexibility later. Another complication is that the approach this study recommends is one that places a premium on design at a time when fashion calls for rapid prototyping, which is often viewed as the antithesis of emphasizing design. My own view is that such extreme versions of rapid prototyping are licenses to steal. Further, experience suggests that a moderate amount of initial design goes a very long way, whereas failure to have any formal design initially reduces substantially the likelihood of achieving well-integrated and seamless results later.

Figure 20 summarizes recommendations on this matter. In this approach, work begins with an initial design, one taking a matter of days or weeks, not months or years. The focus is on the big picture, which translates into defining the decompositional issues well, anticipating variable-resolution needs, building in stubs and sketching out the various trees (as in Fig. 19), and getting all the names straight. In doing this, we must make choices, because a hierarchy that is natural in one application domain may or may not be natural in another. For example, the GEFRAM model described earlier develops a measure of overall force ratio that combines effects of air and ground forces. That has proven quite useful for some applications. By contrast, it might seem quite unnatural in a war gaming simulation used for education, operational planning, or analysis informed by operational considerations.

Having made a first set of choices and designs, the next step is indeed rapid prototyping, focusing on inputs and outputs and obtaining enough insights to permit iteration of the design based on initial simulation results. After some iteration, the structure of the model may well settle down, at which point we can work out details of algorithms and relationships, including the appropriate aggregation relationships and calibration methods. Iteration should continue, but there should be a major effort not to undercut the overall architecture or we will quickly generate numerous seams.

Is the approach a panacea? By no means. It has proven useful, however, in several quite different applications. Further, it seems to have a fair degree of generality that has not yet been exploited—because

- Develop initial design, focusing on composition and top-down views.

- Anticipate need for variable resolution. Build in "stubs." Draw "trees." Choose names to clarify hierarchical relationships.

- Make choices of perspective to determine "best" hierarchical structures. Create explicit hierarchy-breaking approximations.

- Use rapid prototyping. Use first-order algorithms. Focus on inputs and outputs. Use theory to tighten calibration relationships.

- Experiment and iterate design.

- Complete top-level design and proceed. Use more serious algorithms.

- Do not lightly assume "simple" aggregation relationships. Derive from theory when possible. Experiment. Iterate.

- Adapt with applications, but don't undercut basic design.

Fig. 20—Recommended Approach to Design

the value of doing so has not previously been emphasized. Also, as noted earlier, the method will require theoretical efforts to develop good approximations that separate phenomena occurring on different scales and that recognize that aggregation relationships are often complex and nonintuitive. There is a great potential for developing powerful and relatively seamless variable-resolution combat models,[2] but a great deal of work has yet to be done if we are to achieve that potential.

---

[2]An alternative view is that the hierarchy approach is doomed to failure because detailed processes so commonly intrude on higher level views of the problem. Examples of the interference of detailed processes at high levels are very familiar to general officers and even civilians experienced in war gaming. Simulated wars literally go one direction or another as a function of details such as the arrival time of critical reserves in a particular sector, or the range advantage enjoyed by particular weapons. The solution, I believe, is in emphasizing highly interactive models that permit users rapidly and flexibly to "turn the knobs and switches" necessary to include or not include detailed processes in a given application, or even in a given portion of a given simulation run.

Appendix A
BACKGROUND

Variable-resolution issues have been discussed among combat modelers for many years, off and on, but the early work did not leave much of a theoretical legacy. I became interested in the mid-1980s when directing early development of the RAND Strategy Assessment System (RSAS), an analytic war gaming system used for global- and theater-level gaming and analysis. We had multiple objectives and associated tensions regarding the "right" level of resolution. Could we have our cake and eat it? I thought so and encouraged designs that would provide alternative levels of resolution. It was an uphill battle to make this happen, however, and recognized techniques for such designs did not seem to exist. Other organizations encountered similar difficulties. Nonetheless, in the course of the work some techniques emerged that seemed to have general value.

The issue reemerged two years ago in a study (Davis and Blumental, 1991) that elaborated the irony in combat modeling that workers at *all* levels tend to believe that their level of resolution is "correct," that those working at lower resolution are either naive or sloppy, and that those working at higher resolution are lost in the weeds. By contrast, physical scientists learn early to appreciate both thermodynamics and quantum statistical mechanics. Can we develop similarly enlightened views for combat modeling and analysis? DARPA agreed to sponsor work on the issue, and the same problems arose as the Defense Modeling and Simulation Office began worrying about such interoperability issues as how to combine models of different resolutions and have the results be meaningful (DMSO, 1992). Further, the Department of Defense's vision of a seamless continuum between models and reality provided even more motivation for taking the subject seriously. All of this led to a think piece (Davis and Huber, 1992), a small preliminary workshop at RAND late in 1991, and a first interdisciplinary conference on the subject in May 1992, at which an earlier version of this study served as an introduction.

Although I make no attempt in this study to survey the academic literature, there is considerable relevant material, much of it dealing with what is called *model abstraction*. Some useful references here include Innis and Rexstad (1983), Zeigler (1984), Courtois (1985),

Fishwick (1988), and Sevinc (1990). Also, anyone considering modeling of complex systems should see Simon (1981), which discusses the prevalence of "nearly decomposable hierarchies" in real-world systems, including living organisms.

Appendix B

DEFINITION OF THE MODELS

The models used in this study are highly simplified representations of ground combat within a corps sector. They are by no means supposed to be realistic. However, for completeness. I define the models in the following paragraphs, specifying their variables, processes, calibrations, and implementation as programs. Importantly, I describe them as I first developed them, before attempting to integrate, because I wanted to be able to illustrate the difficulties of after-the-fact integration.

THE AGGREGATE MODEL

Table B.1 lists and defines the variables of the aggregate model. The minimal inputs are initial values of attacker and defender strengths, A_o and D_o, and values of the attrition coefficients K_a and K_d. A and D

Table B.1

Variables of the Initial Aggregate Model

Variable	Meaning
A	Attacker strength (equivalent divisions, EDs)
D	Defender strength (EDs)
F	Attacker to defender force ratio: $F = A/D$
ALR	Attacker loss rate (fractional loss per unit time): $ALR = \dfrac{dA/dt}{A}$
DLR	Defender loss rate: $DLR = \dfrac{dD/dt}{D}$
RLR	Ratio of (relative) loss rates: $RLR = \dfrac{ALR}{DLR}$

Parameter	Meaning
K_d	Attrition coefficient: kills per day of attacker EDs per ED of defender
K_a	Attrition coefficient: kills per day of defender EDs per ED of attacker

constitute a complete set of dynamic variables, but it is convenient to compute additional variables that are determined by their definitions and the values of A and D. These are the variables F, ALR, DLR, and RLR.

The only process in this trivial model is that of attrition, governed by a Lanchester square law:

$$\frac{dA}{dt} = -K_d D; \qquad \frac{dD}{dt} = -K_a A \qquad (1)$$

It follows from Eq. 1 that

$$\text{ALR} = -\frac{K_d}{F}; \qquad \text{DLR} = -K_a F; \qquad \text{RLR} = \frac{K_d/K_a}{F^2} \qquad (2)$$

Since there are two independent dynamic variables, two calibration conditions are necessary. The first and most important assumption is the 3 to 1 rule, which says that RLR = 1 when F = 3. That is, the breakeven point is at a force ratio of 3 to 1. A battle that begins with such a force ratio will result in the antagonists destroying each other, with neither side ever improving the force ratio. The assumption is reasonable only if the defender has substantial advantages by virtue of such things as prepared defenses and a favorable terrain.

This expression of the 3 to 1 rule implies (see Eq. 2) that the "defender advantage" obeys:

$$K_d/K_a = 9$$

The other calibration assumption determines the scale of attrition rather than the relative attrition rates. The calculations in the text assume $K_d = 0.18$, which means that at a force ratio of 3 the attacker would be losing 6 percent of its strength per unit time (taken to be a day).

The model was implemented as an EXCEL® spreadsheet simulation program. The differential equations were represented crudely by first-order difference equations with one-day time steps.

The patched version of the aggregate model, referred to only briefly in the text, has two additional parameters, type terrain and force size. A process is also added to adjust effective force strengths and/or effective attrition coefficients to reflect type battle.

constitute a complete set of dynamic variables, but it is convenient to compute additional variables that are determined by their definitions and the values of A and D. These are the variables F, ALR, DLR, and RLR.

The only process in this trivial model is that of attrition, governed by a Lanchester square law:

$$\frac{dA}{dt} = -K_dD; \qquad \frac{dD}{dt} = -K_aA \qquad (1)$$

It follows from Eq. 1 that

$$ALR = -\frac{K_d}{F}; \qquad DLR = -K_aF; \qquad RLR = \frac{K_d/K_a}{F^2} \qquad (2)$$

Since there are two independent dynamic variables, two calibration conditions are necessary. The first and most important assumption is the 3 to 1 rule, which says that RLR = 1 when F = 3. That is, the breakeven point is at a force ratio of 3 to 1. A battle that begins with such a force ratio will result in the antagonists destroying each other, with neither side ever improving the force ratio. The assumption is reasonable only if the defender has substantial advantages by virtue of such things as prepared defenses and a favorable terrain.

This expression of the 3 to 1 rule implies (see Eq. 2) that the "defender advantage" obeys:

$$K_d/K_a = 9$$

The other calibration assumption determines the scale of attrition rather than the relative attrition rates. The calculations in the text assume $K_d = 0.18$, which means that at a force ratio of 3 the attacker would be losing 6 percent of its strength per unit time (taken to be a day).

The model was implemented as an EXCEL® spreadsheet simulation program. The differential equations were represented crudely by first-order difference equations with one-day time steps.

The patched version of the aggregate model, referred to only briefly in the text, has two additional parameters, type terrain and force size. A process is also added to adjust effective force strengths and/or effective attrition coefficients to reflect type battle.

THE "DETAILED" MODEL

Table B.2 lists and defines the variables of the "detailed" model. The minimum set of dynamic variables is $\{A_{tot}, D_{tot}\}$, where A_{tot} and D_{tot} are the total attacker and defender strengths. The problem is defined by their initial values and the values of the parameters shown below them in the table. The attacker and defender strengths on the FLOT, A and D, as well as the variables F, ALR, DLR, and RLR can be calculated for convenience or because they represent important intermediate variables of the attrition process. Note that RLR here is the ratio of relative loss rates for FLOT forces, not for total forces.

This model has three processes, one each for attrition, the decision to move reserves to the FLOT, and the decision to quit the battle if losses are excessive. The principal process assumed was that of

Table B.2

Variables of the Detailed Model

Variable	Meaning
A_{tot}	Total attacker strength (EDs) (FLOT forces plus reserves)
D_{tot}	Total defender strength (EDs) (FLOT forces plus reserves)
A	Attacker strength (EDs) on the forward line of troops (FLOT)
D	Defender strength (EDs) on the forward line of troops
F	Attacker to defender force ratio: $F = A / D$
ALR	Attacker loss rate (fractional loss per unit time): $ALR = \dfrac{dA/dt}{A}$
DLR	Defender loss rate: $DLR = \dfrac{dD/dt}{D}$
RLR	Ratio of (relative) loss rates on the FLOT: $RLR = \dfrac{ALR}{DLR}$

Parameter	Meaning
ADF_{min}	Minimum frontage (km) per attacker ED (shoulder-space limit)
ADF_{max}	Maximum frontage (km) per attacker ED
DDF_{min}	Minimum frontage (km) per defender ED
DDF_{max}	Maximum frontage (km) per defender ED
fD_{doc}	Doctrinally preferred fraction of defender forces on the FLOT
type terrain	Type of terrain on which battle is fought (e.g., open, mixed, or rough)
type battle	Type of battle (e.g., assault on prepared defenses, hasty defenses, . . .)
L	Militarily usable frontage (km)
A_{break}, D_{break}	Percent of original force levels at which attacker and defender armies break off battle ("breakpoints")

Lanchester square attrition for those forces on the FLOT. In its simplest form:

$$\frac{dA}{dt} = -K_d D; \quad \frac{dD}{dt} = -K_a A \tag{3}$$

where A and D are strengths on the FLOT. This does not include the changes in A and D due to reserves moving forward.[1] The model assumes that forces on the FLOT remain there, but forces in reserve can be sent to the FLOT. The decision rules for doing so are as follows. The attacker places as much of his strength on the FLOT as is permitted by the constraint ADF_{min}, which represents a "shoulder-space constraint" (e.g., at least 10 km of frontage per equivalent division). The defender has more complex rules:

- Never put more forces on line than are permitted by the shoulder-space constraint DDF_{min} (seldom a consideration).

- So long as the frontage per ED on the FLOT is no worse than DDF_{max}, maintain a doctrinal fraction fD_{doc} of forces forward.

- If there are too few forces for this, put a larger fraction of the forces forward as necessary to maintain, if possible, frontages no worse than DDF_{max}.

- When this is no longer possible, put all forces on line.

The last process involves breaking off the battle when losses are excessive. The battle ends when either or both of the sides have fractional losses in excess of the relevant "breakpoint" thresholds (A_{break} and D_{break}). Most calculations shown in the test assume there is no breakpoint.

The calculations shown in this study assume baseline parameter values as follows:

K_d = 0.27 (3/2 the value of the corresponding parameter of the aggregate model, since, nominally, only 2/3 of the forces are on line)

ADF_{min} = DDF_{min} = 10 km/ED

---

[1] In non-standard terrains or types of battle, Eq. 3 is modified with numerical multipliers on the right side. For example, in open terrain, the defender would have a smaller advantage than in "normal" terrain, so the multiplier would be less than 1. None of these corrections were used in this study, but their existence is indicated by the parameters type terrain and type battle. In Fig. 16, the parameter "terr" is the same as type terrain in the original detailed model.

$DDF_{max} = 40 \text{ km/ED}$

$fD_{doc} = 0.667$ (i.e., two up and one back).

Again, these models and calibrations are merely illustrative and should not be taken too seriously. This study is about modeling theory, not about combat phenomena.

As with the simple model, implementation involved a simple spreadsheet simulation with one-day time intervals. At the start of each day the new force level on the front was set to the previous day's initial value minus that day's attrition plus whatever reserves would be sent forward consistent with the decision rules discussed earlier. No effort was made to investigate the effects of variable time step, higher precision, etc.

THE REFERENCE MODEL

As discussed in the text, if we want to integrate existing models it is important to develop a comprehensible and complete set of variable names. Table B.3 suggests the variables of a reference model that could embrace all the content of the aggregate and detailed models described here. Read it from the center column. Thus, from the first row, the variable A of the reference model corresponds to A of the aggregate model and A_{tot} of the detailed model. Brackets indicate intermediate variables that were implicit in the models, but important conceptually. For example, the strength of attacker reserves, A_{res}, may have been represented simply as $A_{tot} - A$, but deserves to be given its own name.

Table B.3

Mapping of Variable Names Among Aggregate, Reference, and Detailed Models

Variables of Aggregate Model	Variables of Reference Model	Variables of Detailed Model
A	A	A_{tot}
D	D	D_{tot}
[A']		
[D']		
	A_{flot}	A
	D_{flot}	D
	A_{flot}'	[A']
	D_{flot}'	[D']
	A_{res}	$[A_{res}]$
	D_{res}	$[D_{res}]$
F	F_{tot}	
ALR	ALR_{tot}	
DLR	DLR_{tot}	
RLR	RLR_{tot}	
	ALR	ALR
	DLR	DLR
	RLR	RLR
K_d	K_{dtot}	
K_a	K_{atot}	
	K_d	K_d
	K_a	K_a
type terrain	Type terrain	
force size	Force size	
(K_a')		
(K_d')		
	ADF_{min}, ADF_{max},	ADF_{min}, ADF_{max},
	DDF_{min}, DDF_{max}	DDF_{min}, DDF_{max}
	fD_{doc}	fD_{doc}
	fA_{doc}	
	L	L
	Lg	
	A_{break}, D_{break}	A_{break}, D_{break}
	terr	type terrain
	type battle	type battle

BIBLIOGRAPHY

Allen, Patrick D. (1992), *Situational Force Scoring: Accounting for Combined Arms Effects in Aggregate Combat Models*, RAND, N-3423-NA.

Bennett, Bruce W., Carl M. Jones, Arthur M. Bullock, and Paul K. Davis (1988), *Main Theater Warfare Modeling in the RAND Strategy Assessment System (3.0)*, RAND, N-2743-NA.

Courtois, P. J., (1985), "On Time and Space Decomposition of Complex Structures," *Communications of the ACM*, Vol. 28, No. 6, June, pp. 590–603.

Davis, Paul K. (1987), "A New Analytic Technique for the Study of Deterrence, Escalation Control, and War Termination," in Stephen J. Cimbala, *Artificial Intelligence and National Security*, Lexington Books, D.C. Heath and Company, Lexington, Massachusetts.

Davis, Paul K. (1990), "Central Region Stability at Low Force Levels," in Reiner Huber, *Military Stability: Prerequisites and Analysis Requirements for Conventional Stability in Europe*, Nomos Verlagsgesellschaft, Baden-Baden.

Davis, Paul K., and John J. Arquilla (1991), *Deterring or Coercing Opponents in Crisis: Lessons from the War with Saddam Hussein*, RAND, R-4111-JS.

Davis, Paul K., and Donald Blumenthal (1991), *The Base-of-Sand Problem: A White Paper on the State of Military Combat Modeling*, RAND, N-3148-OSD/DARPA.

Davis, Paul K., Steven C. Bankes, and James P. Kahan (1986), *A New Methodology for Modeling National Command Level Decision-making in War Games and Simulations*, RAND, R-3290-NA.

Davis, Paul K., and Reiner Huber (1992), *Variable-Resolution Combat Modeling: Motivations, Issues, and Principles*, RAND, N-3400-DARPA.

Defense Modeling and Simulation Office (1992), *Defense Modeling and Simulation Initiative*, Office of the Director, Defense Research and Engineering, Department of Defense.

Dupuy, Trevor N. (1987), *Understanding War*, Paragon House Publishers, New York.

48

Epstein, Joshua (1990), *Conventional Force Reductions: A Dynamic Analysis,* The Brookings Institution, Washington, D.C.

Fishwick, Paul A. (1988), The Role of Process Abstraction in Simulation," *IEEE Transactions on Systems, Man, and Cybernetics,* Vol. 18, No. 1, Jan./Feb.

Hillestad, Richard, and Mario Juncosa (forthcoming), *Cutting Some Trees to See the Forest: On Aggregation and Disaggregation in Combat Models,* RAND, R-4250-DARPA, forthcoming.

Hillestad, Richard, John Owen, and Donald Blumenthal (forthcoming), *Experiments in Variable Resolution Combat Modeling,* RAND, N-3631-DARPA, forthcoming.

Horrigan, Timothy (1991), *Configural Theory and the Mathematical Modeling of Combat: Realizing the Potential of Models and Simulations of Combat to Improve the Effectiveness of Weapons, Tactics, and Training,* Horrigan Analytics, HAS 91-17-1, Chicago, Illinois.

Innis, G., and E. Rexstad (1983), "Simulation Model Simplification Techniques," *Simulation,* July, pp. 7–15.

Sevinc, Suleyman (1990), "Automation of Simplification in Discrete Event Modelling and Simulation," in *Int. J. General Systems,* Vol. 18, pp. 125–145.

Simon, Herbert (1981), *Sciences of the Artificial,* 2nd ed., MIT Press, Cambridge, Massachusetts.

Zeigler, Bernard (1984), *Multifaceted Modelling and Discrete Event Simulation,* Academic Press, London.